WICE USB 使用手冊

義隆電子系統應用二處編著

初版

4/1/2004

目　　錄

一、介紹 3
二、如何開始 3
三、組譯與連結 22
1. 組譯與連結處理流程 22
2. 指令規則(Statement Syntax) 22
3. 數字資料型態表示 23
4. 組譯時的數學運算(Assembler arithmetic operations) 23
5. 程式指示(Program directives) 25
6. 條件式組譯(Conditional assembly) 30
7. 保留字(Reserved word) 32
四、除錯(debugger) 33
1. 設定中斷點(Set breakpoint) 33
2. 執行程式(run program) 33
五、WICE 環境介紹 35
1. 選項介紹(Menu) 35
2. 工具列(Tool bar) 36
3. 連接 (connect) 36
4. 檔案功能(File menu) 37
5. 編輯功能(Edit menu) 43
6. 顯示功能(View menu) 44
7. 計畫功能(Project menu) 47
8. 除錯功能(Debug menu) 50
9. 工具功能(Tool menu) 54
10. 選擇性的功能(Option menu) 55
11. 視窗(Window) 58
12. 求助(Help) 60
六、錯誤訊息(error message) 61
七、問題與討論(Q&A) 71

一、介紹

WICE USB 是一套義隆 8 位元微控制器 Window 版 USB 介面 ICE 整合發展環境軟體（包括編輯、計畫管理(Project)、組譯、除錯），可執行於 Window 95 及 Window 98, Window Me, Window 2000 及 Window XP 作業系統。WICE USB 主要的目標在於讓使用者方便使用、支援更多功能以及增強傳輸及程式的穩定性。

介紹 WICE USB 主要分為以下七個部份：

一、介紹

二、如何開始

三、組譯（assembler）與連結(linker)

四、除錯（debugger）

五、USB WICE 環境介紹

六、錯誤訊息(error message)

七、問題與討論

WICE USB 是一套以計畫(Project)導向的整合發展環境軟體,整合了編輯、計畫管理(Project manager)、組譯及除錯，因此在使用上更加方便，功能上提供多種選擇，如 Editor 加入尋找(find)、取代(replace)、undo、 redo、剪下(cut)、拷貝(copy)及貼上(paste)，發展環境以計畫為導向，使用原始程式除錯及組譯器加入插入檔案(include)、巨集(macro)、組譯時數學運算(assembly arithmetic)、條件式組譯(conditional assembly)、list file 及 map file 等。

除了基本功能外，WICE USB 還加入了讓使用者方便操作的功能，如區間註解(block comment)、直接修改暫存器、直接反組譯的功能及增加與 ICE 傳輸速度等。WICE USB 努力的目標就是提供使用者更方便的使用環境、更好用的功能及更快速更穩定的 ICE 環境。

二、如何開始

一般使用者拿到一個新軟體首先第一個問題是如何開始操作，而 WICE USB 是計畫導向的 ICE 整合發展環境，因此本章節就是教使用者如何開始建立一個計畫(Project)，如何增加一個新檔案到計畫中，或者增加已存在的檔案到計畫，然後如何組譯或建立一個機械碼(CDS code)，接者如何使用基本除錯功能。

Step 1:安裝 ICE USB Software

1.1 將壓縮檔案雙擊後，會解壓縮成兩個部分（一為 USB ICE 主程式部分，另一為驅動程式部分）；使用滑鼠雙擊主程式部分，就會進入安裝畫面（如圖 1）。完成後，點選下一步『NEXT』。

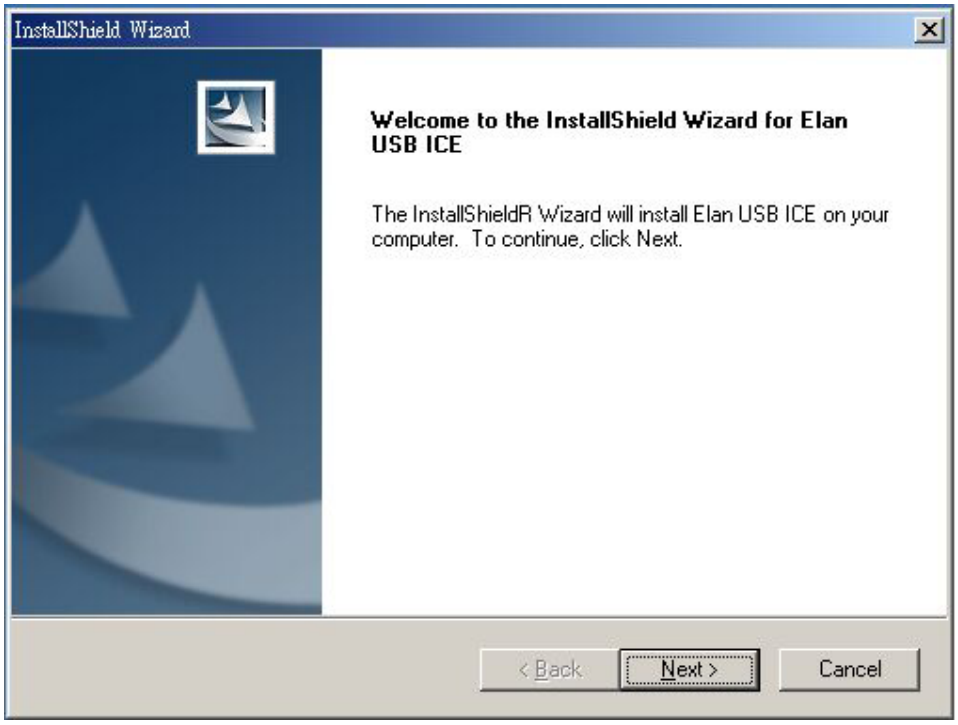


圖 1：安裝程式歡迎畫面

1.2 此時點選下一步『NEXT』，繼續安裝。繼續圖 2 的版權宣告頁面，仔細閱讀後，選擇接受『ACCEPT』版權宣告的內容，繼續軟體安裝；若此時選擇不接受，『NO』則會退出安裝程式。完成後，點選下一步『NEXT』。

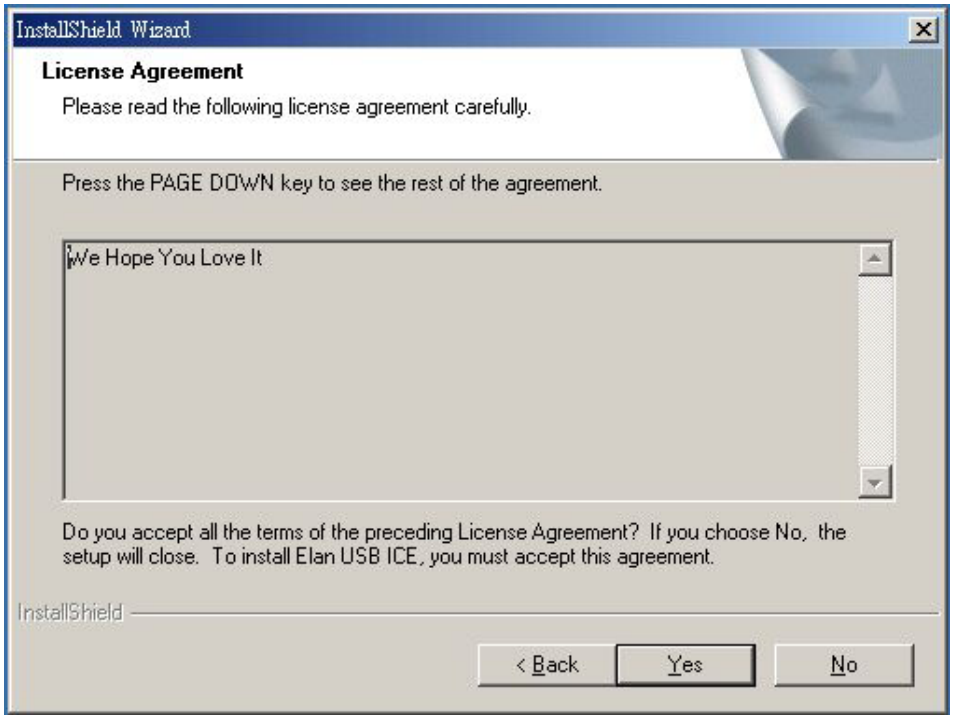


圖 2：版權宣告（選擇接受後繼續，否則會退出安裝）

1.3. 接下來是一些介紹此 WICE 的相關訊息，如圖 3。完成後，點選下一步『NEXT』。

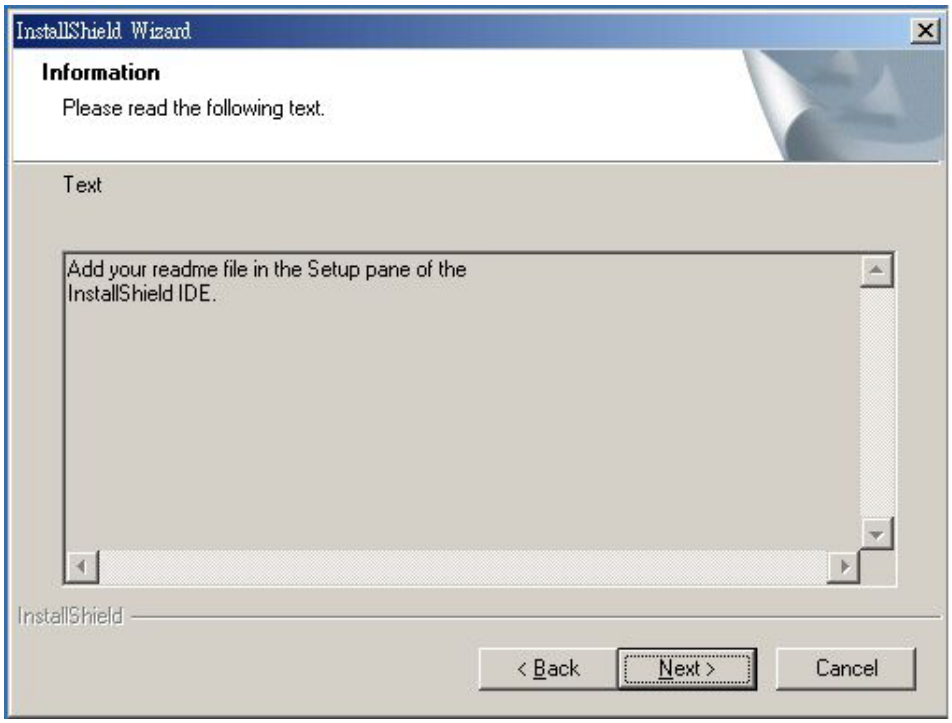


圖 3：軟體資訊

1.4. 請輸入使用者的相關訊息（使用者姓名及組織名稱），如圖 4。完成後，點選下一步『NEXT』。

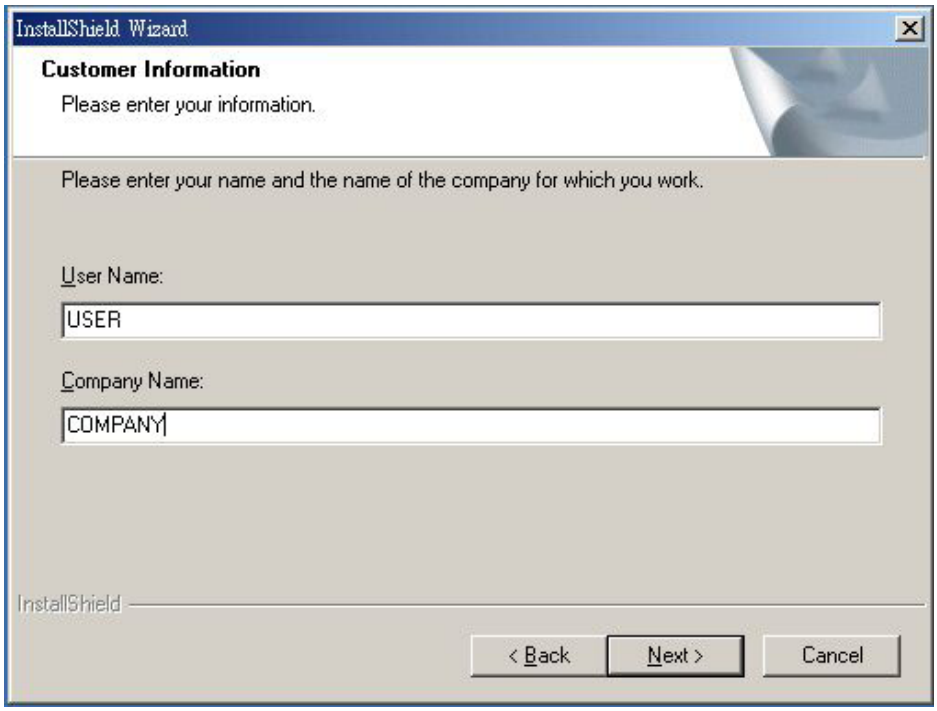


圖 4：使用者及組織相關訊息

1.5 選擇安裝模式（如圖 5），可選擇典型安裝『Typical』；若硬碟空間不足，請選擇最小安裝『Compact』；亦可依自己喜好選擇自訂安裝『Custom』。若無特殊需要，強烈建議以『典型安裝』進行程式安裝流程。完成後，點選下一步『NEXT』。

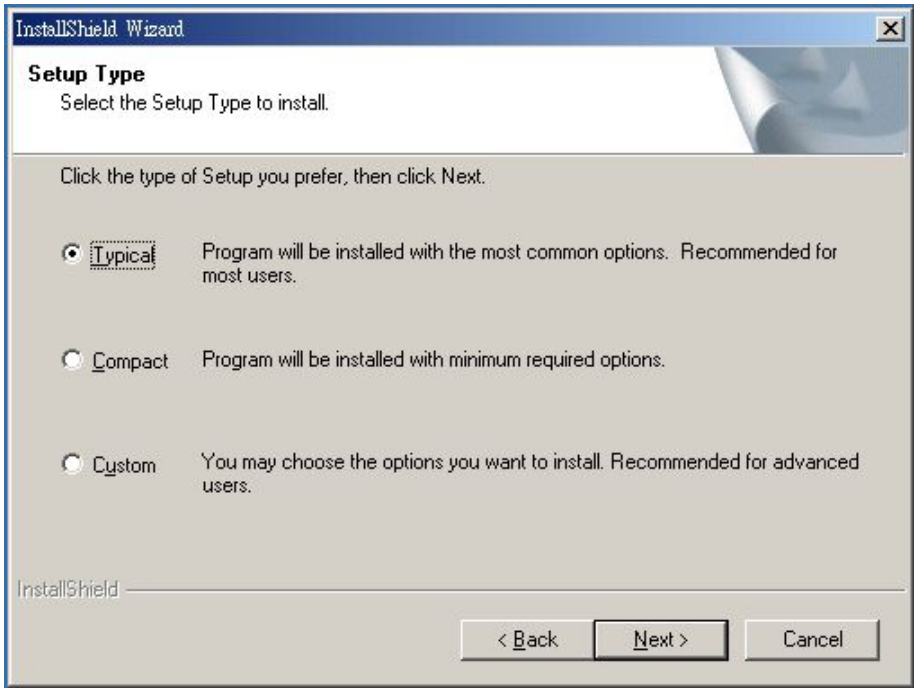


圖 5：安裝流程選項

1.6 選擇程式顯示名稱及安裝位置（預設值為 Elan USB ICE 1.01；安裝時可能因版本不同所顯示的數字不盡相同）。完成後，點選下一步『NEXT』。

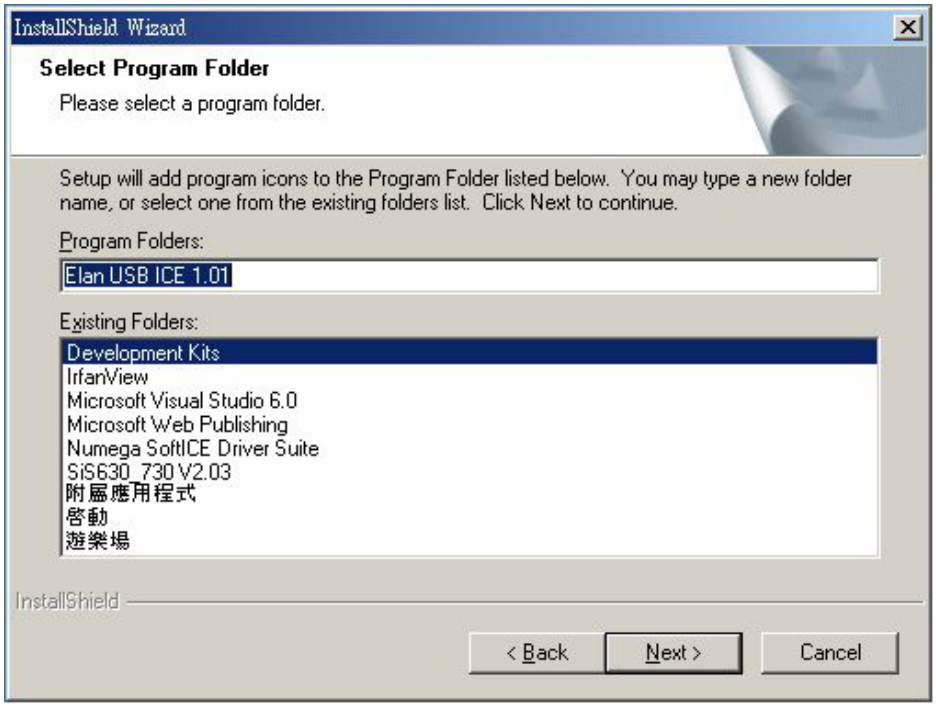


圖 6：選擇安裝目錄

1.7.開始複製需要的檔案至對應系統目錄。完成後，點選下一步『NEXT』。

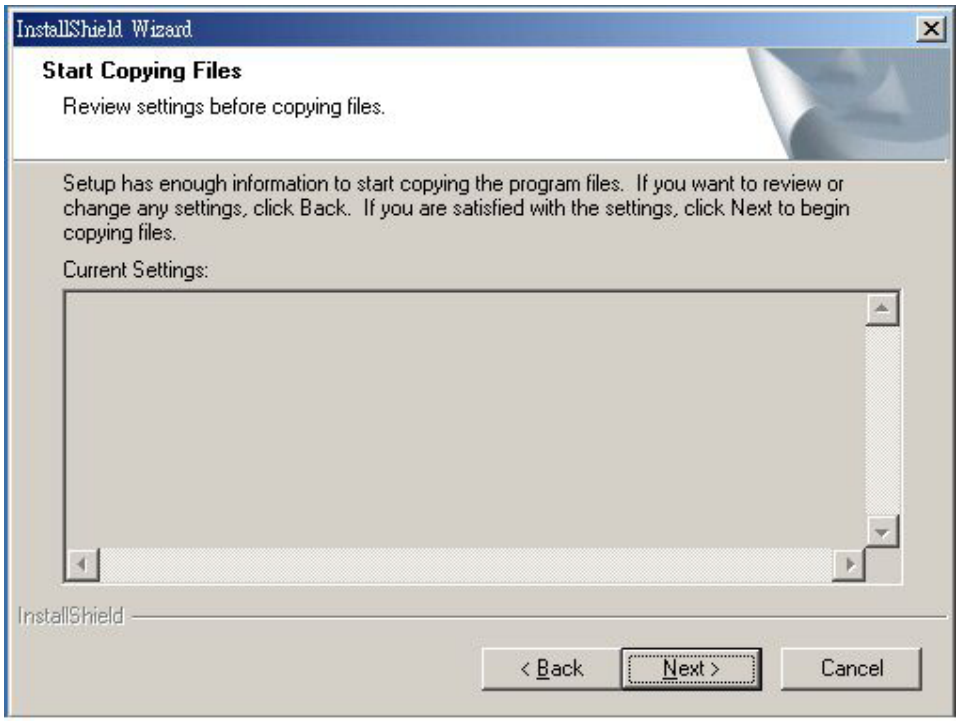


圖 7：複製檔案，準備開始安裝

1.8.複製檔案完成度百分比，等到安裝進度到達 100%，點選下一步『NEXT』。

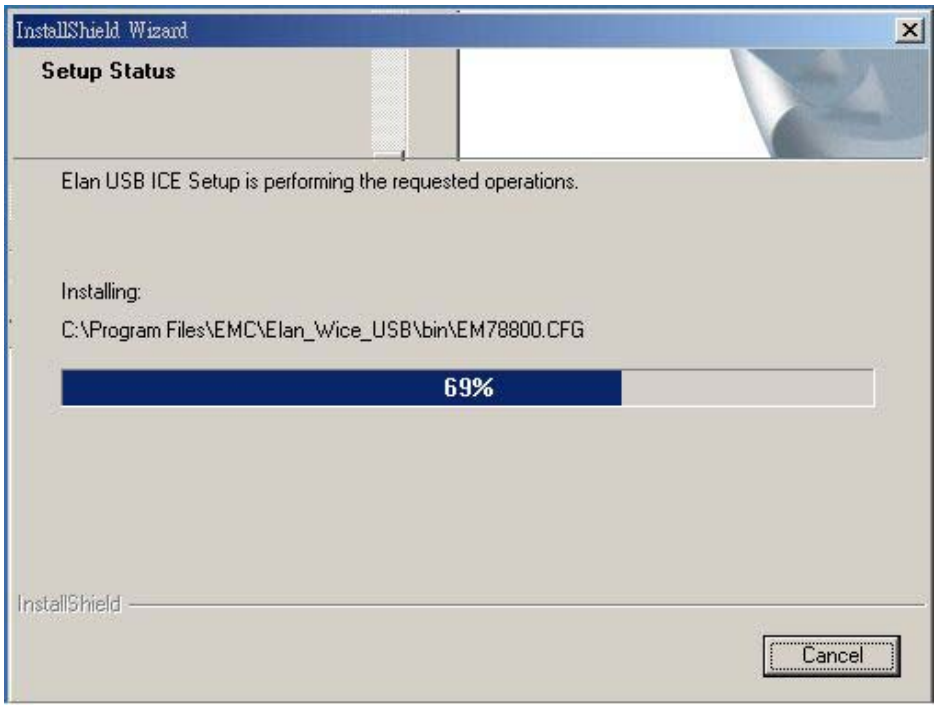


圖 8：安裝進度

1.9.安裝完成，按『Finish』後關閉安裝程式。

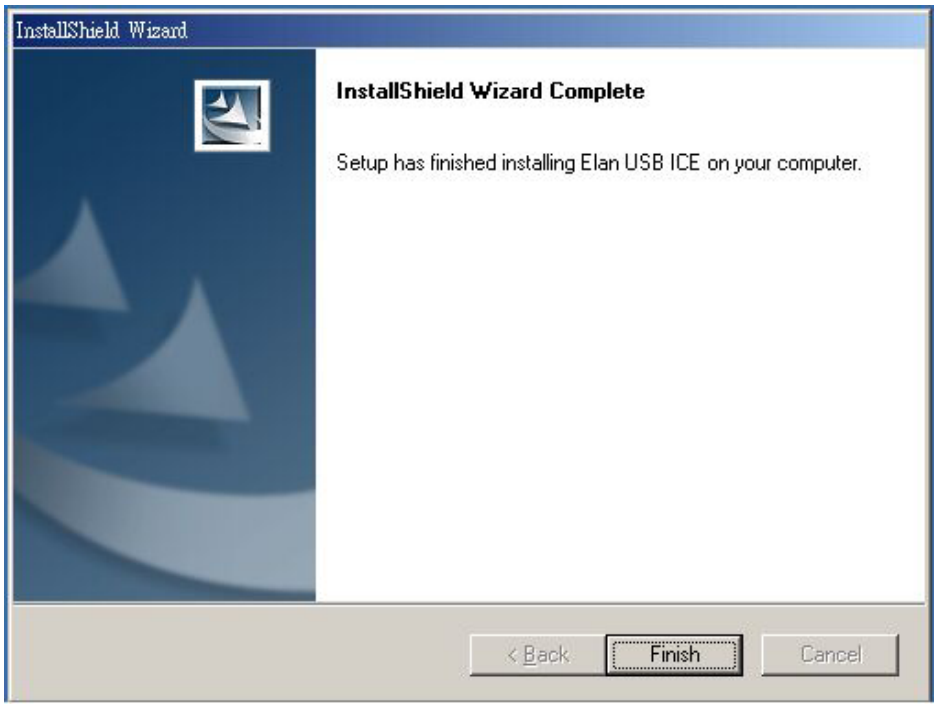


圖 9：完成安裝

Step 2:硬體安裝

將 USB ICE 插入空的 USB 連接埠，此時會在桌面彈跳出一個安裝驅動程式的視窗，可直接進行 Step 3。

在 USB ICE 面板上有兩個 LED 燈，在上方的是 RUN LED，當燈亮時表示程式正在執行；在下方的是 POWER LED，當燈亮時，表示 USB ICE 是上電狀態(因為此 USB ICE 裝置是 Bus-power 供電模式，因此不需要外接電源供電)。

在面板右上方有一個 6*2 的插座，是用來插入適用的外掛震盪器(如 6MHz，12MHz 震盪器)，插入時，注意方向性及對應腳位即可。

圖 10：系統裝置接線圖

Step 3:安裝驅動程式

3.1 **Step1** 安裝程式完成後，若安裝過程無誤，系統會在桌面產生一個 WICE USB 捷徑圖示（若無此圖示，請重複 **Step1**，再一次安裝）

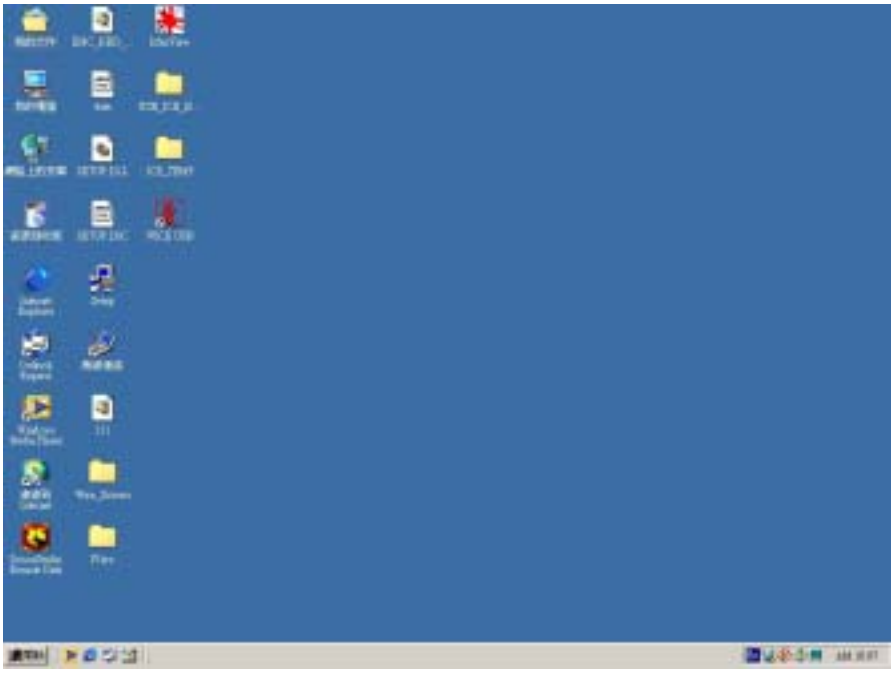


圖 11：安裝完成後會在桌面新增一捷徑

3.1 (Drivers 相關檔案可至 Elan Corp.官方網站下載；網址為http://www.emc.com.tw 中文版/ 產品櫥窗/ 電腦周邊設備 IC/ USB Controller/ Supporting Tools/ Wice (USB)) 完成後，透過 USB 連接線將 USB ICE 裝置與 PC 連接，此時系統因無法識別 USB ICE 裝置，故彈跳出一個『新增硬體視窗』，要求使用者選擇對應的驅動程式，此時選擇『指定驅動程式的位置（進階）(S)』，按『下一步』繼續。

圖 12：指定安裝驅動程式

3.3 指定『驅動程式』位置（使用『瀏覽』選定驅動程式所在位置；作業系統為 WinXP 或 Win2000 者，請選擇\\Driver\WinXP2000 目錄，若為 Win98 或 WinME 者，請選擇 \\Driver\Win98 目錄），選定完成後按『下一步』繼續。

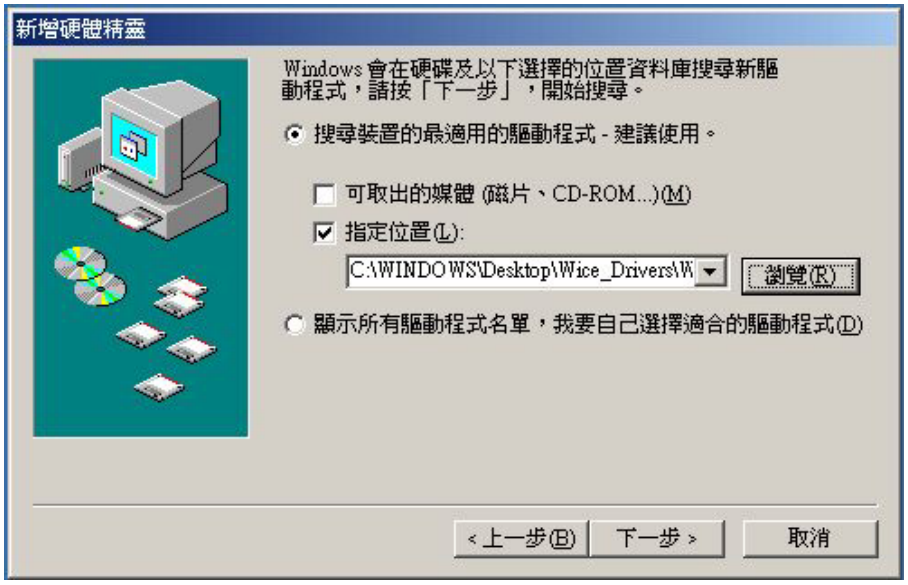


圖 13：指定驅動程式所在位置（選擇先前解壓縮後『Driver』目錄所在位置）

3.4 找到對應的驅動程式後，點選『下一步』繼續。

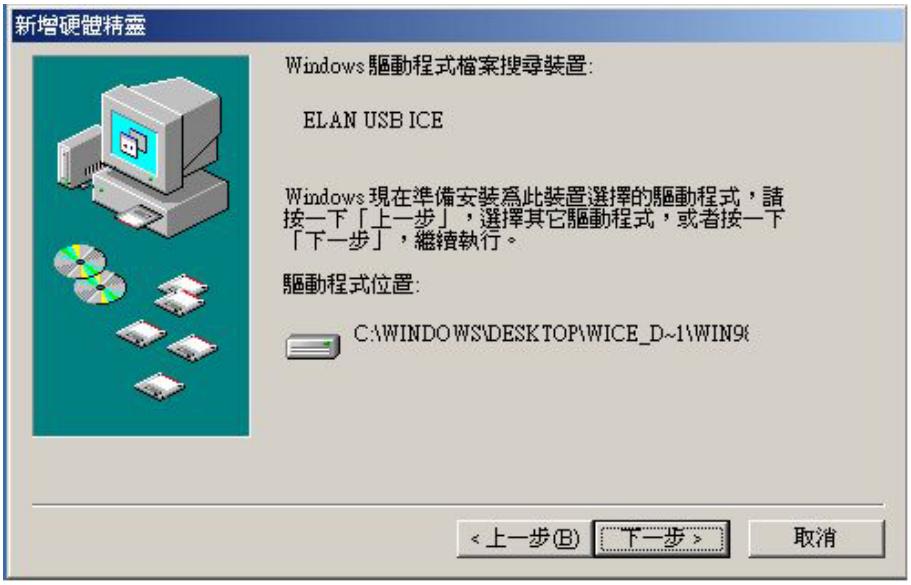


圖 14：根據驅動程式，系統可成功識別 ELAN USB ICE

3.5 驅動程式安裝完成後，新增的硬體即可正常使用。按下『完成』結束安裝。

圖 15：驅動程式安裝完成，USB ICE 可正常使用

此時可藉『開始』→『設定』→『控制台』→系統→硬體 頁中的『裝置管理員』查詢得知 ELAN USB ICE 已正常運作。

圖 16：ELAN USB ICE 可正常運作

Step 4: 進入 WICE

方法一:可以雙擊鍵(double click)桌面的 WICE USB 的圖示(ICON)。

方法二:選擇[開始]→[程式集] →[EMC WICE] →[WICE USB]

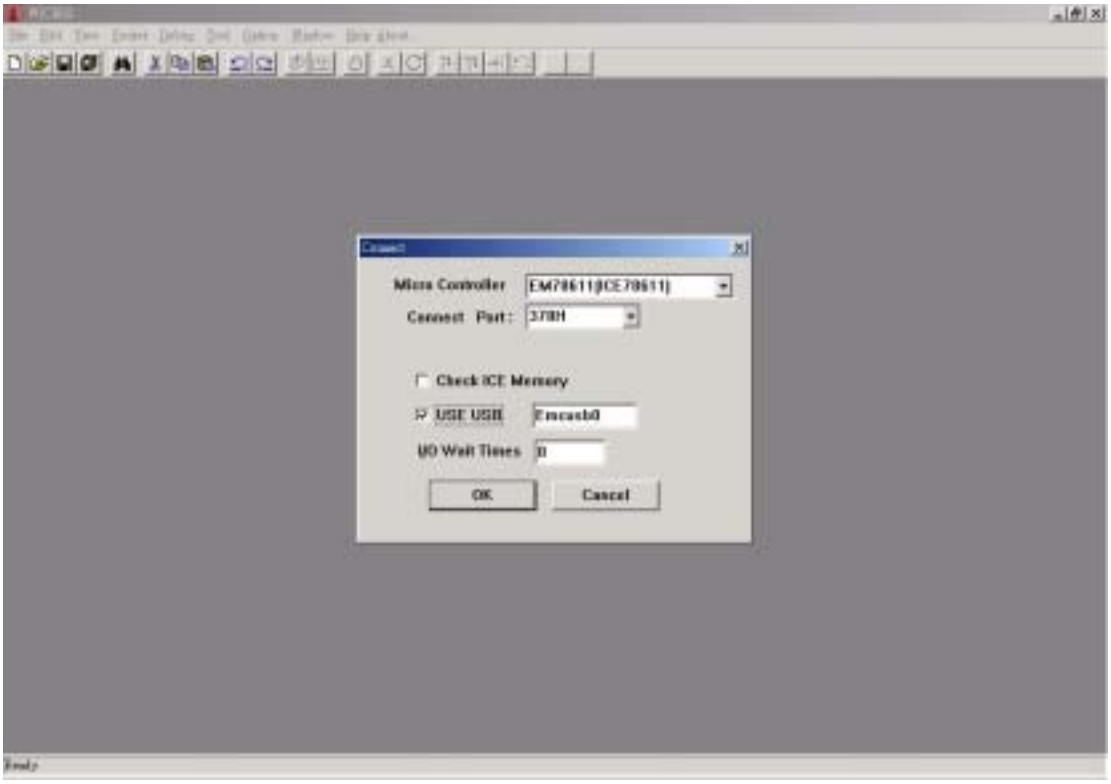


圖 17:起始畫面

Step 5: Connect

1. Micro Controller:請選擇對應的微控制器名稱，如 EM78611(ICE78611)；如此，WICE USB 才能根據對應的微控制器作出正確的組態設定。
2. Connect Port:此選項是平行序列埠選項設定，保留預設值即可。
3. Check ICE Memory:接後，接著檢查 ICE 程式記憶體是否有錯，以確定程式記憶體是否正常運作(此選項可依使用者習慣勾選與否)。
4. USE USB:勾選後，系統會試著檢查 USB ICE 是否透過 USB 線連接到 USB ICE，如果有，會在檢查無誤後連接成功；如果此時沒有連接 USB ICE，則會產生連接失敗(error U0001: Open USB Driver Error)訊息(此選項為必須勾選選項)。
5. 按下 OK 鍵後，系統會依您選擇的項目進行與 WICE USB 連結。

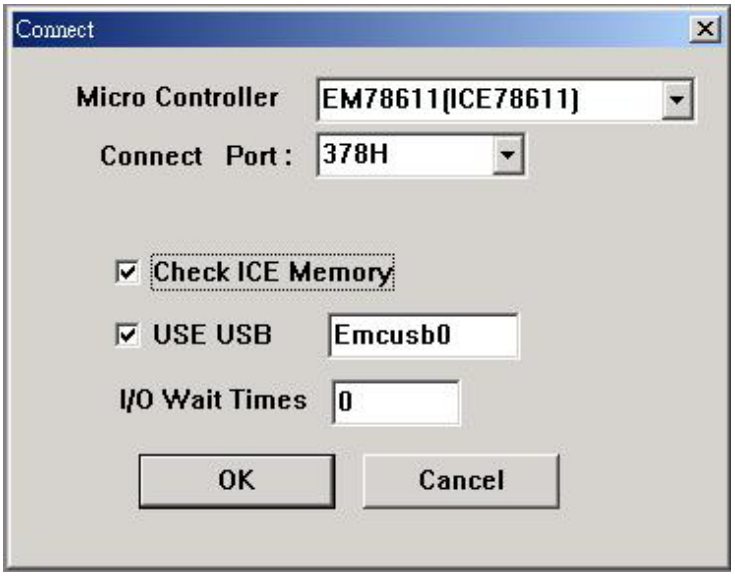


圖 18:Connect 畫面

Step 6: Code Option 畫面設定:可設定的選項為

1. Frequency:可供選擇的頻率為 12MHz 與 6MHz，請依附掛的振盪器選擇適當的選項。
2. OST1, OST0:此為設定振盪器起振時間，組合方式有

OST1	OST0	TIME
0	0	500 μs
0	1	2 ms
1	0	8 ms
1	1	16 ms

3. D- Resistor:低速裝置(Low Speed Device)附掛電阻方式，可選擇內掛(IC 內建電阻致能)或是選用外掛電阻(內部附掛電阻禁能)。
4. IC Select:選用不同的 IC 編號(目前選項為 MASK 或 MTP)，因不同類型 IC 編號其內部的控制暫存器定義不完全相同，請依對應的微控制器作出適當選擇。

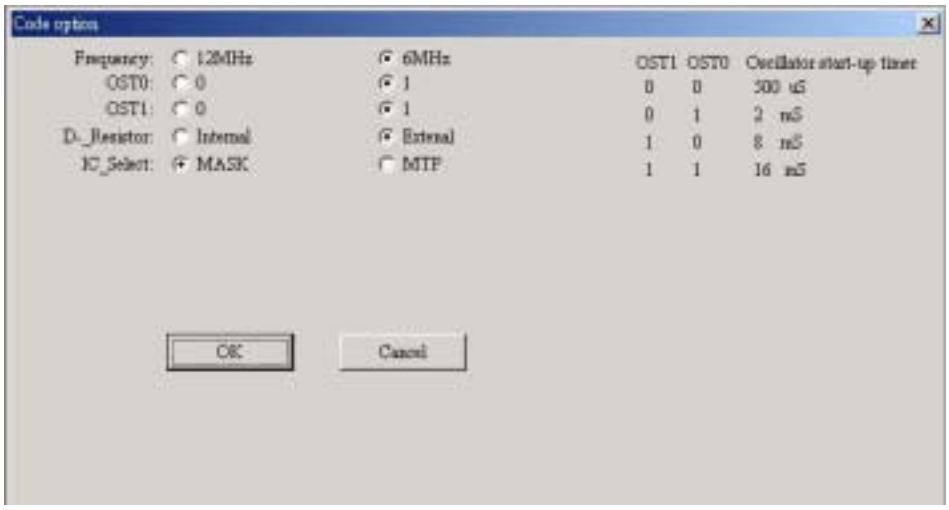


圖 19:Code Option 設定畫面

Step 7: 新增一個計畫(Project):

1. 選擇[File]→[New…], 顯示新增的對話盒(dialog)。
2. 選擇[Project] tab.

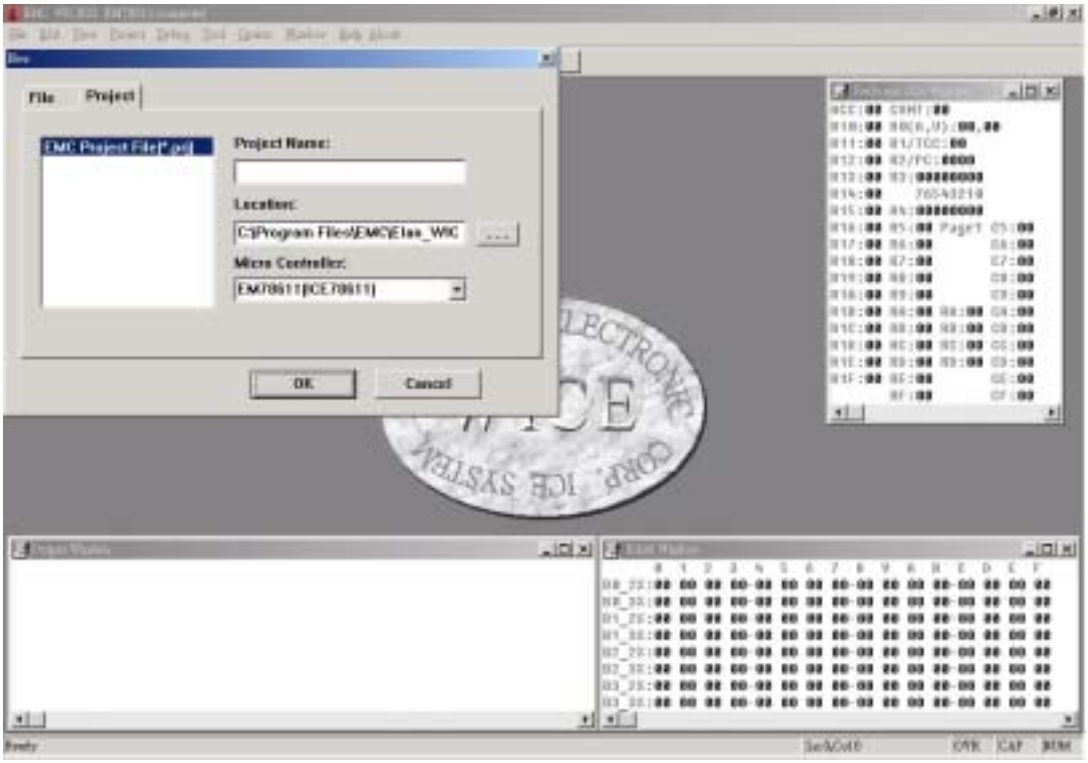


圖 20 從[File]→[New]新增一個 Project

也可選擇[Project] → [New…]

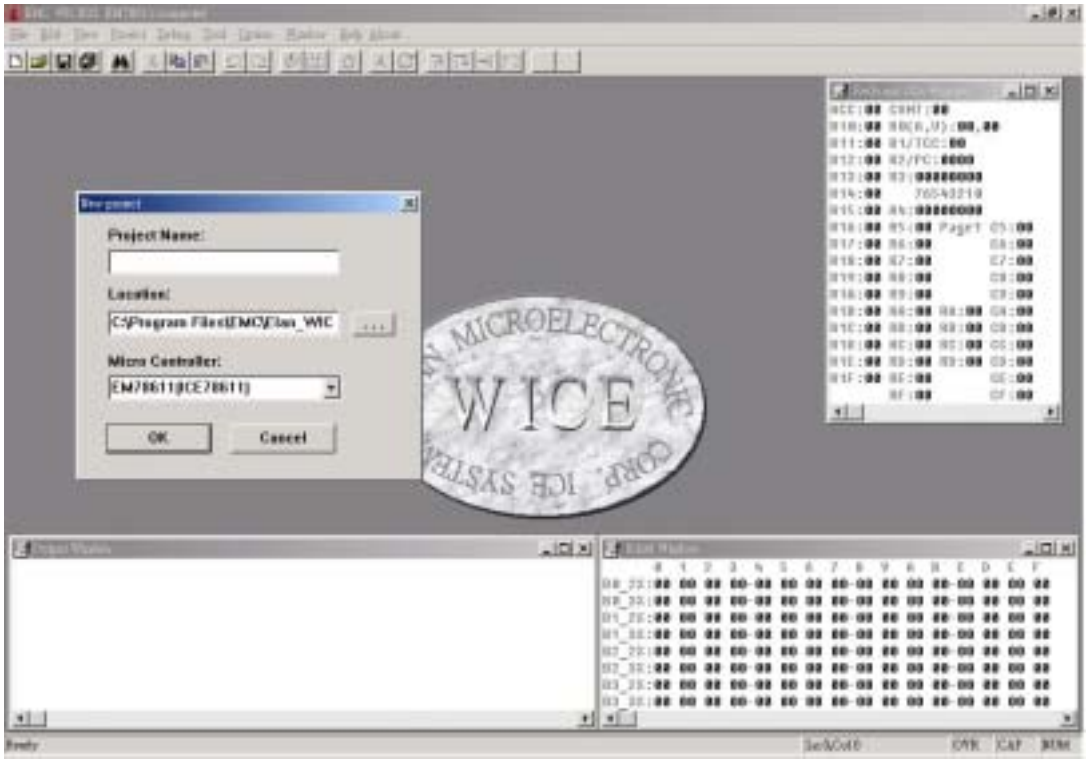


圖 21:從[Project]→[New] 新增一個 Project

Step 8: 新增一個新檔案到計畫(假如沒有已編輯的檔案)

1. 選擇[File]→[New…], 顯示新增的對話盒(dialog)。
2. 選擇[File] tab。

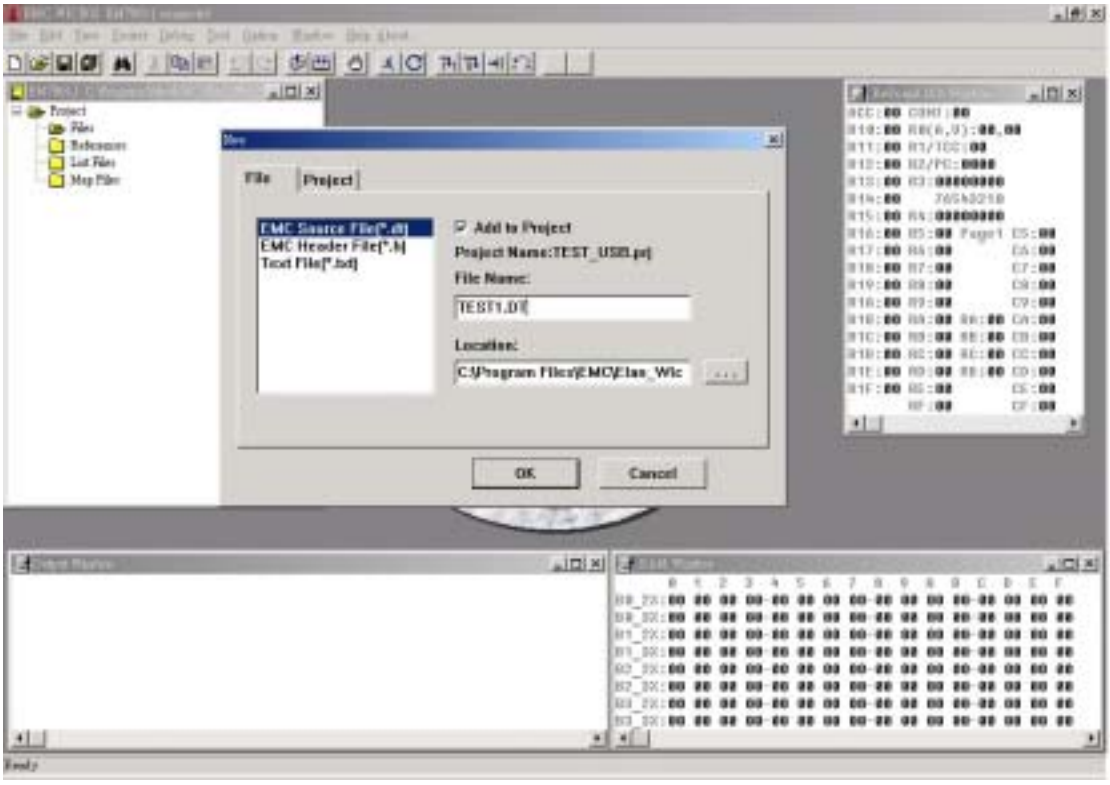


圖 22:新增一個檔案

Step 9: 新增一個舊檔案到計畫(假如已經有編輯好的檔案)

1. 選擇 [Project] →[Add files to project…], 顯示 open file 對話盒 (dialog)。
2. 選擇欲加入 project 的檔案，並按 OK 按鍵。

圖 23:增加一個舊檔到 Project

Step 10: 編輯程式

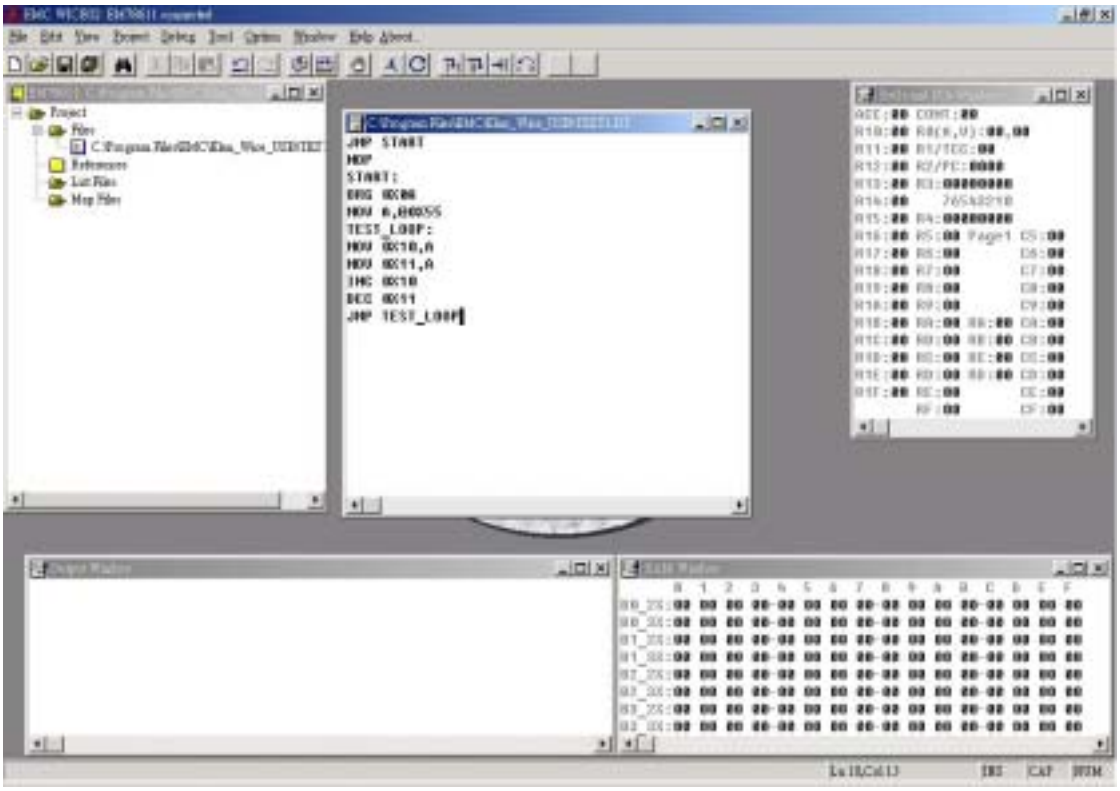


圖 24:編輯程式

Step 11: 組譯並連結

選擇[Project] →[Rebuild All], Output 視窗會顯示組譯及連結的訊息。

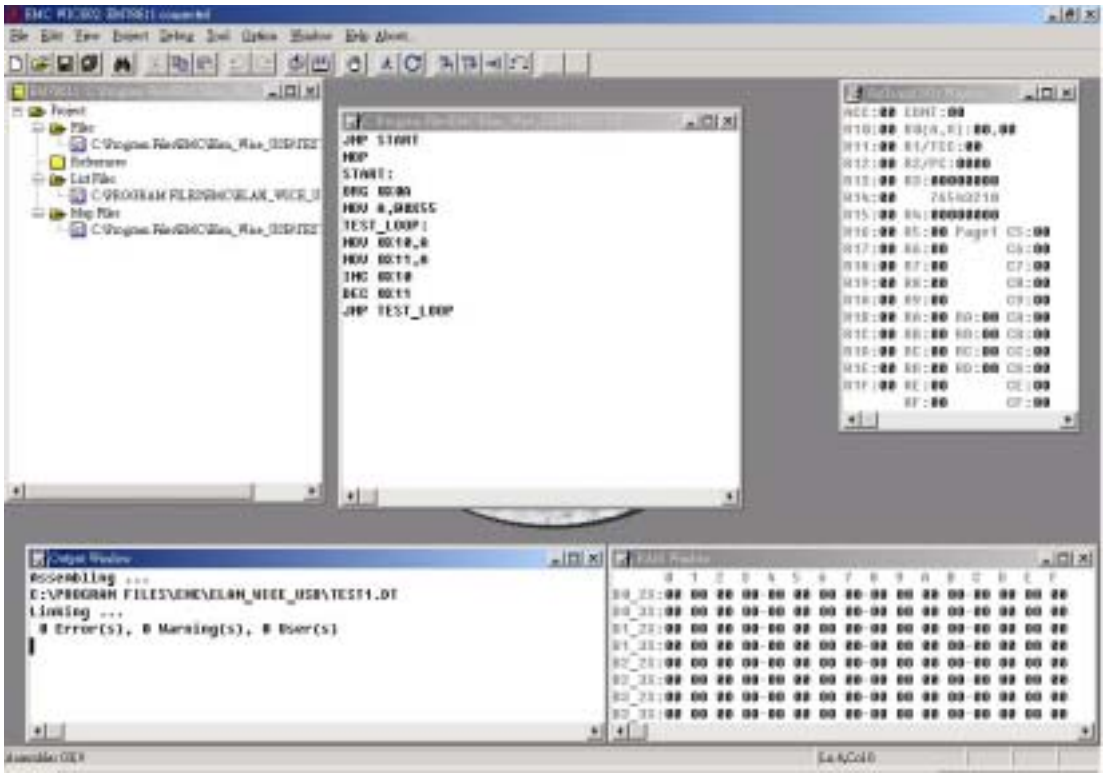


圖 25: 組譯及連結

Step 12: 將程式載入 ICE

選擇[Project] →[Dump to ICE]

圖 26: 將程式載入 ICE

Step 13: 除錯(Step Into)

選擇[Debug] →[Step Into] 或選擇熱鍵(Hot key) [F7]

Step 14: 設中斷

將游標移到需中斷的來源檔(副檔名 dt 或 asm)，用滑鼠雙擊鍵(Double click).

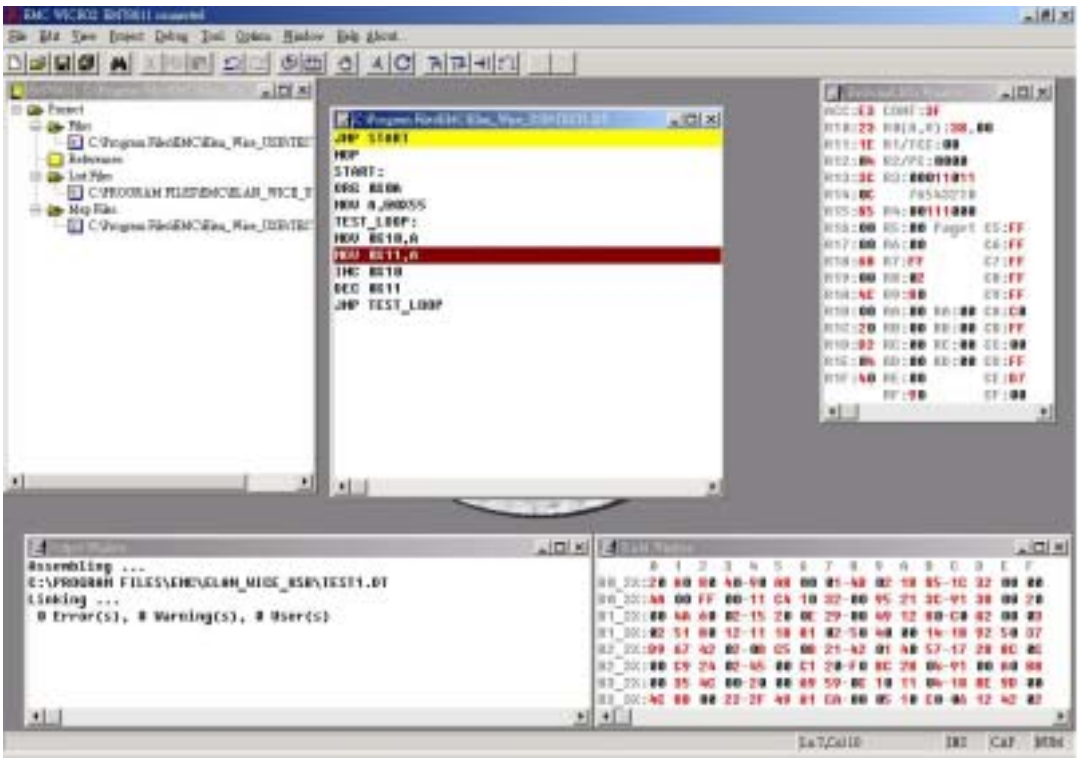


圖 27:設定中斷點

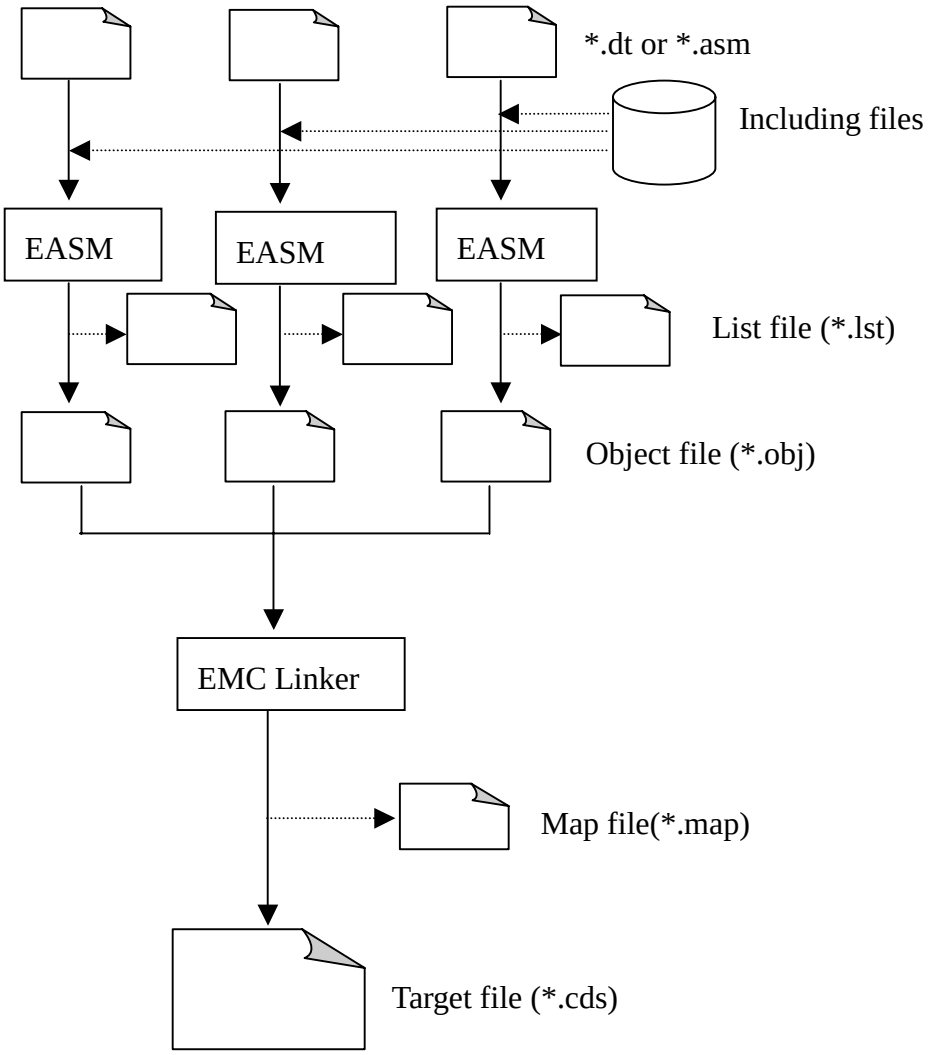
Step 15: 執行 Go

選擇[Debug] →[Go]或選擇 Hot key [F5]

Step 16: 重覆 step 10 至 step 15，直到程式正確為止

三、組譯與連結

1. 組譯與連結處理流程



2. 指令規則(Statement Syntax)

[label [:]] operation [operand] [,operand][; comment]

中括弧代表括弧內的內容可有可無，且標記(label)，運算子(operation)，運算元(operand) 和註解(comment)內的字元均無大小寫區分，而各單元(如標記，運算子)是以空白或跳格鍵(Tab)區隔。

標記(label): 標記後的冒號（:）可有可無，但建議程式設計者應該加冒號，在可讀性上較為清楚，若標記後接巨集執行指令，此標記一定需要加上冒號，以區隔標記與巨

集。標記的定義限制，只能使用大寫及小寫英文字母，數字鍵及底線，也就是 A~Z，a~z，0~9 和_，0~9 數字不能當標記第一個字母且標記長度最多 31 個字。

運算子(operation): 包括指令（instruction）及指示(directive)，指令是指可組譯成機械碼的文字，如 mov a,@02，而指示(directive)是指不組譯機械碼的文字，主要是方便程式設計者更方便程式設計，如 org 0x20。

運算元(operand): 運算元的數目與特性主要是依據運算子(operation)而定，也就是提供運算子處理的變數，運算元與運算元之間是以逗號（,）區隔。

註解(comment):註解的意義是讓使用者解釋某些程式的意義，方便程式設計者日後方便程式修改及維護。註解分為二種，行註解(line comment), 區塊註解(block comment)。

行註解:是以分號（;）為起頭的第一字元，在分號之後到下一行為止的所有文字描述均為註解，例如 ; test comment

區塊註解(block comment): 是以倒斜線加星號（/*）及星號加斜線（*/），作為註解的前後，中間所有文字描述均為註解，因此可包括多行的文字註解，例如/* test comment */

3. 數字資料型態表示

資料型態	表示方法一	表示方法二	表示方法三
十進制(Decimal)	0D<digits>	<digits>D	<digits>
十六進制(Hexadecimal)	0X<digits>	<digits>H	
八進制(Octal)	0Q<digits>	<digits>Q	
二進制(Binary)	0B<digits>	<digits>B	

註: 十六進制中若以<digit>H 表示時，若數字第一個字是 A~F 或 a~f 則需在數字前加個 0。若表示方法一與方法二衝突, 則已表示方法一為優先考量。

4. 組譯時的數學運算(Assembler arithmetic operations)

組譯時的數學運算是表示所有運算式在組譯時均都必須能計算結果，否則會產生錯誤，每一個計算均需為整數，若為浮點數時，則系統自動轉為整數處理。

真(TRUE) 和假(FALSE)：以 0xFF 表示真，以 0x00 表示假。

以下所歸納的運算子(operator)是以計算優先順序排列。

(1) 左括弧 (和右括弧)

(2) 只需一個運算元的運算子(Unary operators)

! 非(Logical NOT)

~ 補數(Complement)

－ 減(Unary minus)

(3) 乘、除、餘數及移位

* 乘(Multiplication)

/ 除(Division)

% 餘數(Modulo)

<< 左移(Logical shift left)

>> 右移(Logical shift right)

(4) 加、減

＋ 加

－ 減

(5) 位元 AND

& (BIT AND)

(6) 位元 OR 及 XOR

| 位元 OR (Bit OR)

^ 位元 XOR (Bit XOR)

(7) 邏輯 AND

&& 邏輯 AND (Logical AND)

(8) 邏輯 OR

|| 邏輯 OR (Logical OR)

(9) 比較(Comparison)

== 等於(equal)

!= 不等於(not equal)

> 大於(greater than)

< 小於(less than)

>= 大於等於(greater or equal than)

<= 小於等於(less or equal than)

5. 程式指示(Program directives)

(1) ORG : 設定程式計數器(program counter)的值。

語法: ORG <expression>

例如: org 0x200

(2) EQU or == (Duplicate =): 常數設定

語法: <label> EQU <expression>

例一: R20 equ 0x20

例二: R20 == 0x20

(3) 註解

行註解(Line comment)

語法: ; < string >

例如: ; this is the comment string.

區間註解(Block comment)

語法: /* <strings> */

例如: /* this is block comment

example including multi lines */

(4) EOP : 指程式 ROM Page 的結束(end of page)。

語法: EOP

例如:

```
        org     0x10
        mov     0x20,A
        inc     0x20
        eop
        inc     0x20
```

組譯成下列的結果(source 前為 address):

```
        org     0x10
0010    mov     0x20,A
0011    inc     0x20
        eop
```

```
0400    inc         0x20
```

(5) END :程式結束,在 END 之後的程式均不組譯。

語法: END

例如:

```
        org         0x10
        mov         0x20,a
        inc         0x20
        end
        mov         0x20,a
```

組譯成下列的結果(source 前為 address):

```
        org         0x10
0010    mov         0x20,a
0011    inc         0x20
        end
        mov         0x20,a
```

(6) PROC, ENDP : 副程式的定義，增加副程式的可讀性。

語法: <label> PROC

```
            <statements>
                ENDP
```

例如:

```
    BANK0:    PROC
              BC     0X04,6
              BC     0X04,7
              RET
              ENDP
```

註:PROC 及 ENDP 只是增加副程式的可讀性，所以 RET 在副程式還是必須的.

(7) INCLUDE: 插入別的檔案，讓程式更為簡潔及維護。

INCLUDE:分為兩種，一種系統提供的 INCLUDE 檔案，另一種是使用者自己撰寫的 INCLUDE 檔案。

a. 使用系統提供的 INCLUDE 檔案，如 EMC456.INC，EMC32.INC

語法: INCLUDE <filename>

例如: INCLUDE <EMC456.INC>

b. 使用使用者自定的 INCLUDE 檔案。

語法: INCLUDE "file_path+file_name"

例如: INCLUDE "C:\EMC\TEST\TEST456.INC"

需將路徑完整寫出，否則將無法找到檔案。

註:一般使用 INCLUDE 檔通常都寫變數及巨集定義，以增加擴充性及可讀性。

(8) PUBLIC 及 EXTERN:宣告變數為內部程式定義 或外部程式定義 因本軟體採用 Project 導向設計使用者程式，故同一 Project，不同檔案間的互相呼叫需使用 PUBLIC 及 EXTERN，讓組譯器能夠辨認此變數屬於內部程式或外部程式。

PUBLIC 語法: PUBLIC <label>[,<label>]

EXTERN 語法: EXTERN <label>[,<label>]

PUBLIC 及 EXTERN 可在程式任何地方宣告，也可多個 PUBLIC 及 EXTERN。

例如:在一個 Project 內包含二個檔案，TEST1.DT 及 TEST2.DT

TEST1.DT:

```
        org     0x00
                public  start
                extern  loop1
Start:          mov     a,@0x02
                mov     0x20,a
                jmp     loop1
```

TEST2.DT:

```
                org     0x100
                public  loop1
            extern  start
Loop1:          inc     0x20
```

```
                jmp     start
```

在 TEST1.DT 中因為 START 標籤是內部程式所定義且為外部程式使用，而 LOOP1 為外部程式所定義且為內部程式使用，所以 START 須定義為 PUBLIC，而 LOOP1 則定義為 EXTERN。

在 TEST2.DT 則恰好相反，因此 START 須定義為 EXTERN，而 LOOP1 則須定義為 PUBLIC。

(9) VAR:組譯時期的變數，此變數的值只能在組譯時改變.

語法: Label VAR <expression>

例如:

```
test    var     1
mov     a,@test
test    var     test+1
mov     a,@test
```

(10) MACRO, ENDM:巨集定義指令

語法: <label> MACRO <parameters>

```
  statements
ENDM
```

例如:

```
bank0   macro
    bc  0x04,6
    bc  0x04,7
    endm
```

(11) MACEXIT: 跳離巨集指令，遇到 MACEXIT 則離開巨集，其餘巨集內的指令均不組譯。

語法: MACEXIT

例如:

Source:

```
test  var   5
bank0       macro
            bc  0x04,6
```

```
        if      test>4
        macexit
        endif
            bc    0x04,7
        endm
    bank0
```

組譯後(前為 address)

```
0000 bc    0x04,6
```

因 test=5 所以 if test>4 符合條件，遇到 macexit，所以一直到 endm 的所有程式均不組譯。

(12) MESSAGE :在 output 視窗顯示使用者自行撰寫的字串。

語法:

```
MESSAGE   “<characters>”
```

例如:

```
org 0x00
message “ set bank to 0 !!”
bc    0x04,6
bc    0x04,7
```

組譯後在 output 視窗顯示:

USER MESSAGE: set bank to 0!!

(13) $: 目前程式計數器(Program counter)的值

將 $ 當做運算元(operand)使用。

例如:

```
    jmp     $
```

jmp $ 是指 jmp 同一行，當做死回圈使用。

6. 條件式組譯(Conditional assembly)

(1) IF : IF 條件式組譯，假如 IF 後面的運算式的值為真，則執行 IF 下一行的程式碼，直到遇到 ELSEIF 或 ELSE 或 ENDIF。

語法: IF <expr>
<statements1>
[ELSEIF <expr>
<statements2>]
[ELSE
<statements3>]
ENDIF

例如:

```
org 0x00
bank  macro  num
 if num==0
    bc   0x04,6
    bc   0x04,7
 elseif num==1
    bs   0x04,6
    bc   0x04,7
 elseif num==2
    bc   0x04,6
    bs   0x04,7
 elseif num==3
    bs   0x04,6
    bs   0x04,7
 else
    message “error : bank num over max number!!!”
 endif
endm
```

(2) IFE: IFE 條件式組譯，假如 IFE 後面的運算式的值為假，則執行 IFE 下一行的程式碼，直到遇到 ELSEIFE 或 ELSE 或 ENDIF。

語法: IFE <expr>
<statements1>
[ELSEIFE <expr>
<statements2>]
[ELSE
<statements3>]
ENDIF

(3) IFDEF: IFDEF 條件式組譯，假如 IFDEF 後面的標簽(label)已定義，則執行 IFDEF 下一行的程式碼，直到遇到 ELSEIFDEF 或 ELSE 或 ENDIF。

語法: IFDEF <label>
<statements1>
[ELSEIFDEF <label>
<statements2>]
[ELSE
<statements3>]
ENDIF

例如:

```
org 0x00
ice456  equ  456
ifdef  ice456
    bc  0x04,6
    bc  0x04,7
endif
```

(4) IFNDEF: IFNDEF 條件式組譯, 假如 IFNDEF 後面的標簽(label)未定義, 則執行 IFNDEF 下一行的程式碼，直到遇到 ELSEIFNDEF 或 ELSE 或 ENDIF。

語法: IFNDEF <label>
<statements1>

[ELSEIFNDEF <label>

<statements2>]

[ELSE

<statements3>]

ENDIF

7. 保留字(Reserved word)

(1) 指示(directives),運算子(operators)

+	-	*	/	==
!=	$	@	#	(
)	!	~	%	<<
>>	&	|	^	&&
||	<	<=	>	>=
DS	ELSE	ELSEIF	ELSEIFDEF	ELSEIFE
ELSEIFNDEF	END	ENDIF	ENDM	ENDMOD
ENDP	EQU	EXTERN	IF	IFE
IFDEF	IFNDEF	INCLUDE	MACRO	MACEXIT
MODULE	NOP	PAGE	ORG	PROC
PUBLIC				

(2) 指令集(instructions mnemonics)

ADD	AND	BC	BS	CALL
CLR	COM	COMA	CONTR	CONTW
DAA	DEC	DECA	DISI	DJZ
DJZA	ENI	INC	INCA	INT
IOR	IOW	JBC	JBS	JMP
JZ	JZA	MOV	NOP	OR
RET	RETI	RETL	RLC	RLCA
RRC	RRCA	SLEP	SUB	SWAP
TBL	WDTC	XOR		

(3) 暫存器名稱(Register names)

A

四、除錯(debugger)

本軟體提供的除錯是屬於來源除錯(source level debugger)，除錯的處理步驟:

Step 1:設定中斷點(若將執行 go 需先設定中斷點，其餘執行方式均不用設定)。

Step 2:執行程式，包括了 go，free run，reset，step into，step over，go to cursor。

1. 設定中斷點(Set breakpoint)

(1)使用中斷預設計數值(breakpoint counter): 所謂中斷計數值是指執行程式經過此中斷點幾次會產生中斷。

Step1: 選擇[option]→[Debug option setting]則顯示對話盒(dialog)，設定[Default break point counter]的值(大於 1 小於 255)。

Step2: 將游標移到欲產生中斷的位置，然後雙擊鍵(double click)滑鼠右鍵，則會產生一行棕色的線條，代表中斷產生成功。產生中斷方式也可使用 tool bar 上的 toggle breakpoint (手的形狀)，或使用 menu 上[Debug] →[Toggle Breakpoint]。

(2)設定中斷時才輸入次數

Step1: 選擇[option]→[Debug option setting]則顯示對話盒(dialog)，設定[Default break point counter]的值為 0。

Step2: 將游標移到欲產生中斷的位置，然後雙擊鍵(double click)滑鼠右鍵，則會產生一個輸入中斷次數的對話盒(dialog)，輸入次數後按對話盒的[OK]，則出現一行棕色的線條，代表中斷產生成功。

2. 執行程式(run program)

每次執行完程式則會讀取程式計數器，暫存器的值，以供程式設計者除錯之用。中括弧內為 hot key。

(1) go(F5):從目前的程式計數器(program counter)開始執行到中斷發生，會將中斷的那一行程式執行完畢。

(2) free run(F10): 從目前的程式計數器(program counter)開始執行，忽略所有的中斷，直到使用者取消執行。

(3) reset(F6): 執行 ICE 的 reset 動作，將程式計數器回到 reset 的初始值。

(4) step into(F7):單步執行，每次執行一行指令，讓程式設計者作為細部除錯。

(5) step over(F8):遇到 call 指令則會將整個副程式一次執行完畢，直到遇到 RET 指令為止，若為其它指令則單步執行(step into)。

(6) go to cursor(F4): 從目前的程式計數器(program counter)開始執行到目前游標所在的位置。

五、WICE 環境介紹

1. 選項介紹(Menu)

圖 28:選項(Menu)

(1) File:檔案及計畫的處理，檔案處理包括開啟新檔、開啟舊檔、儲存檔案、另存新檔及關閉檔案，計畫(Project)處理包括開啟計畫、儲存計畫及關閉計畫，列印檔案，儲存最近關閉的檔案及計畫。

(2) Edit:編輯處理:Undo，Redo，拷貝(Copy)，貼上(Paste)，剪下(Cut)，至第幾行(go to line)，Find(尋找)，Replace(取代)。

(3) View:查看視窗處理:可選擇使用者想看的視窗、Project、Special Register、RAM(Bank)、Output、Watch。

(4) Project:計畫處理:新增(new), 開啟(open), 儲存(save), 關閉(close), 增加舊檔到計畫(Add files to project)，刪除在計畫內的檔案(Delete files from project), 組譯一個檔案(Assembler file)，組譯未經過組譯的檔案並連結(Build)，全部組譯並連結(Rebuild All)，將程式碼讀入 ICE (Dump to ICE)，查看曾經執行的程式指令(Trace log)。

(5) Debug:除錯處理:設定中斷點, 清除所有中斷點, 執行程式, 執行方式包括 go、 free run、step into、step over、reset、go to cursor、run from selected line、reset and go，reset and free run。

(6) Tool:工具,包括檢查 ICE 記憶體(Check ICE memory)，產生單棵 Piggy back (Piggy back MIX format)，產生兩顆 Piggy back (Piggy back Hi Lo format)，讀取 check sum (Get Check sum from project)。

(7) Option:功能處理:連接(connect), 特殊暫存器的顯示型態(Register view), 除錯設定(Debug option setting), ICE code option 的設定(ICE code option)，環境變數設定(environment setting)，字型設定(Font)。

(8) Window:視窗排列設定，及所有開啟的視窗名稱。

(9) Help:輔助說明。

(10)About: 有關軟體版本及軟體資訊顯示。

2. 工具列(Tool bar)

圖 29: Tool bar

工具列從左至右分別為(1)開啟新檔(New file)，(2)開啟舊檔(Open file)，(3)儲存檔案(Save)，(4). 儲存所有檔案(Save all)，(5)尋找(Find)，(6)剪下(Cut)，(7)拷貝(Copy)，(8)貼上(Paste)，(9)Undo，(10)Redo，(11)組譯(Assembler)，(12)組譯並連結(Build)，(13)設中斷點(Toggle break point)，(14)Go，(15)Reset，(16)Step into，(18)Step over，(19) Go to cursor，(19)Run from selected line，(20)Trace back， (21) Trace forth。

(1)至(4)為檔案管理， (5)至(10)為編輯管理，(11)至(12)為計畫(Project)管理，(13)至(21)為除錯處理。

3. 連接 (connect)

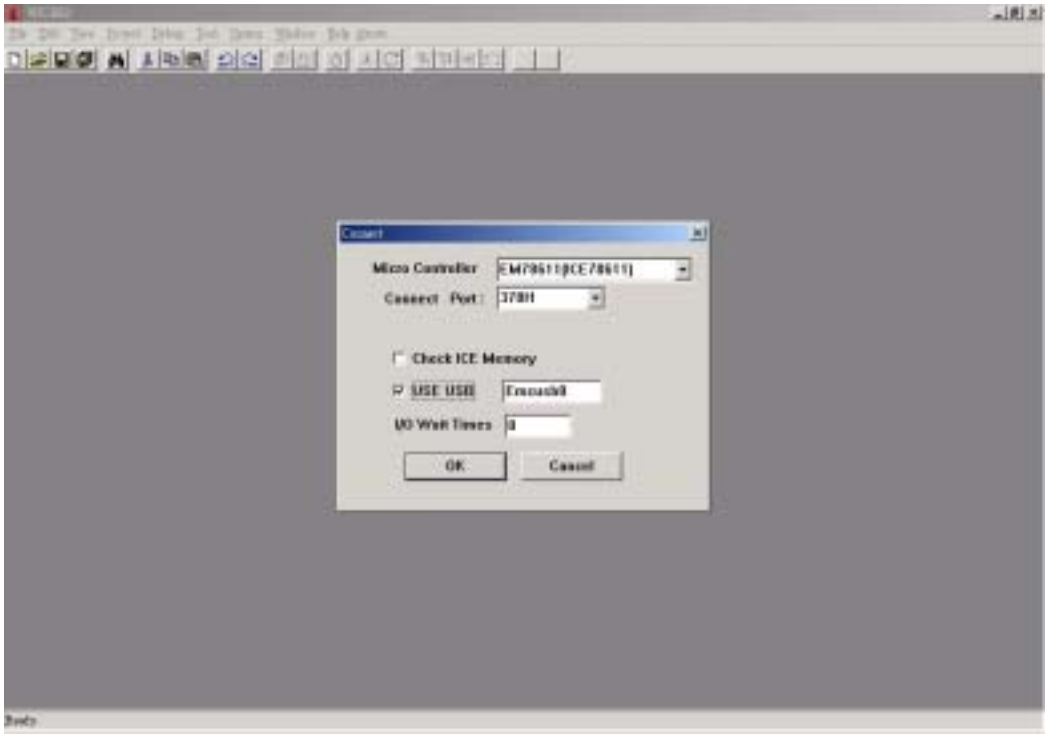


圖 30:連接 USB ICE

Micro controller:義隆八位元微控制器名稱，如 EM78611, EM78612, EM78810, EM78811 等。

Connect port:為 Print port 的位址，如 378H, 278H, 3BCH(因目前使用 USB PORT 連結，故不理

會此項設定，保持預設值即可)。

Check ICE Memory: 連接後，接著檢查 ICE 程式記憶體是否有錯，以確定程式記憶體是否正常運作。

USE USB:勾選後，系統會試著檢查 USB ICE 是否透過 USB 線連接到 USB ICE，如果有，會在檢查無誤後連接成功；如果此時沒有連接 USB ICE，則會產生連接失敗訊息。

重新連接 ICE:選擇[Option]→[Connect], 會出現圖 29 的對話盒，使用者可選擇微控制器名稱，以重新連接，可是若再次連接同樣的微控制器，則會將電腦與 ICE 之間的傳輸等待時間拉長，以解決傳輸協定(protocol)的問題。

4. 檔案功能(File menu)

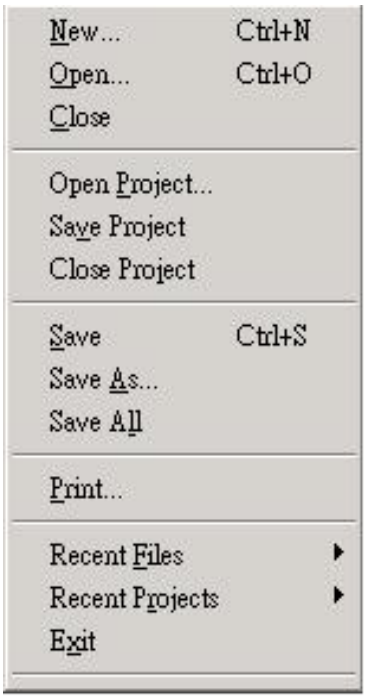


圖 31:檔案功能的所有選項

(1) New (新增檔案或 Project)

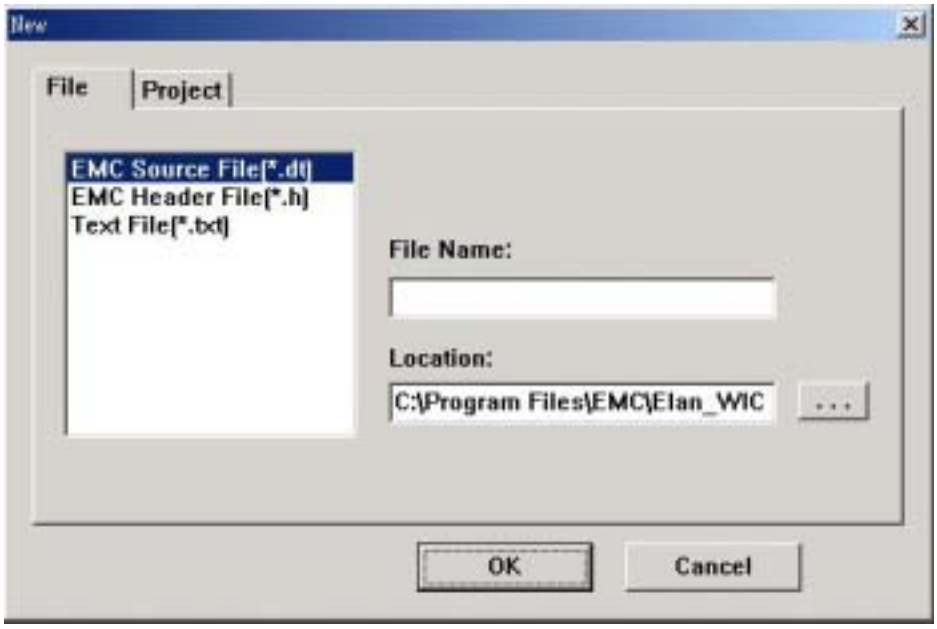


圖 32: 新增檔案

新增檔案(New file):新增一個空白檔案到計畫

step 1:使用者可使用[Add to Project]的檢查盒(check box)的勾選，來決定這個空白檔案是否加入 Project，

step 2:鍵入檔案名稱，在檔案名稱(file name)的下面有一行編輯欄(edit)，可鍵入使用者所希望的檔案名稱。

Step 3:選擇路徑，在檔案路徑(location)的下面有一個路徑名稱，使用者可選擇右邊的三個點的按鈕[…]，按下後選擇所希望的路徑。

Step 4:確定全部選項無誤之後，按下確定(OK)按鈕，則可完成新增一個檔案，或覺得不想新增檔案，可按取消(Cancel)按鈕。

新增一個 Project:新增一個 Project，以開始一個計劃進行。

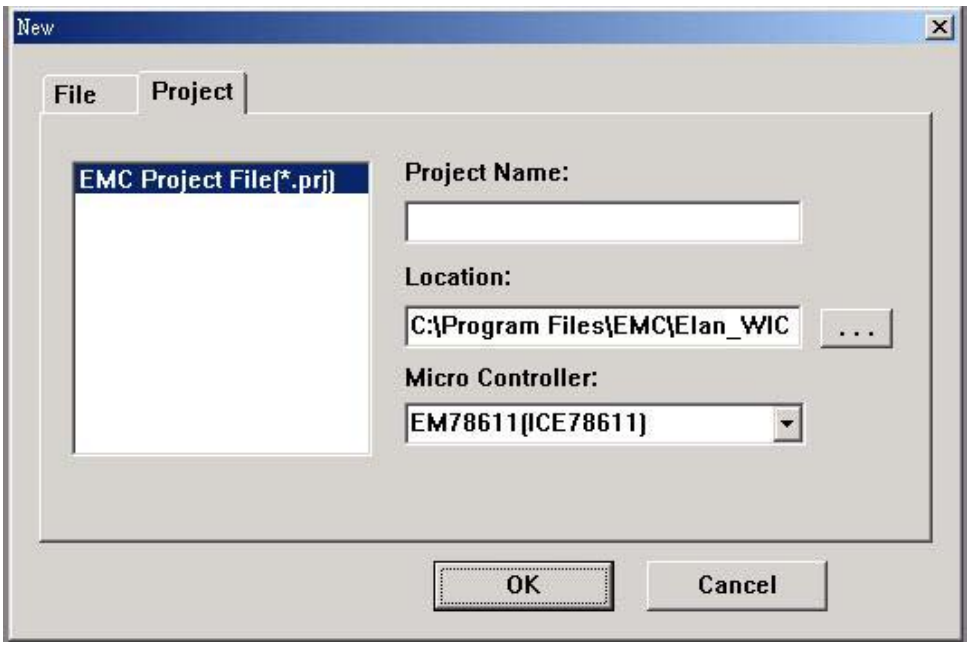


圖 33:新增一個計畫

Step 1:選擇[Project]的標記(tab)。

Step 2: 鍵入計畫(Project)名稱，在計畫名稱(Project name)的下面有一行編輯欄(edit)，可鍵入使用者所希望的計畫名稱。

Step 3:選擇路徑，在路徑(location)的下面有一個路徑名稱，使用者可選擇右邊的三個點的按鈕[…]，按下後選擇所希望的路徑。

Step 4:選擇此計畫所用的微控制器(Micro Controller)，使用拉下視窗選擇微控制器。

Step 5:確定全部選項無誤之後，按下確定(OK)按鈕，則可完成新增一個計畫，或覺得不想新增計畫，可按取消(Cancel)按鈕。

(2) Open (開啟檔案或計畫(project))

圖 34:開啟檔案或計畫(project)

開啟檔案或計畫(project):

Step 1:選擇檔案類型，如來源檔(Source)的副檔名為 dt 或 asm，計畫檔(Project)的副檔名為 prj。

Step 2:選擇欲開啟的檔案或計畫，將滑鼠移到欲選擇的檔案，按下滑鼠左鍵，則檔案名稱會反白。

Step 3:按下開啟舊檔(Open)的按鈕。

(3) Close (關閉檔案或計畫(project))

Step 1: 使用滑鼠選擇欲關閉的檔案或計畫，按下滑鼠的左鍵，使選擇的視窗(window)，變成主要的視窗(Active window)。

Step 2: 選擇[File]→[Close]，則所選的視窗會關閉。

(4) Open Project (開啟計畫(project))

圖 35:開啟計畫

Step 1:使用滑鼠選擇欲開啟的計畫。

Step 2:使用滑鼠按下開啟舊檔的按鈕。

(5) Save Project (儲存計畫(project))

Step 1:使用滑鼠選擇欲儲存的計畫，儲存計畫的內容包括計畫內的檔案名稱(只有名稱不存檔案內容)，包括來源檔(Source)，參考檔(Reference)，List file (組譯後產生的參考檔案)，Map file(連結(link)產生的共用標籤(Public label)的參考檔)及 ROM less Code Option 的值。

Step 2:選擇[File]→[Save Project]，則儲存計畫的內容。

(6) Close Project (關閉計畫(project))

選擇[File]→[Close Project]，則關閉目前在 WICE 的計畫視窗(Project Window)。

(7) Save (儲存檔案或計畫(project))

Step 1: 使用滑鼠選擇欲儲存的檔案或計畫，按下滑鼠的左鍵，使選擇的視窗(window)，變成主要的視窗(Active window)。

Step 2: 選擇[File]→[Save]，則所選的視窗(檔案或計畫)會儲存。

(8) Save As (另存新檔)

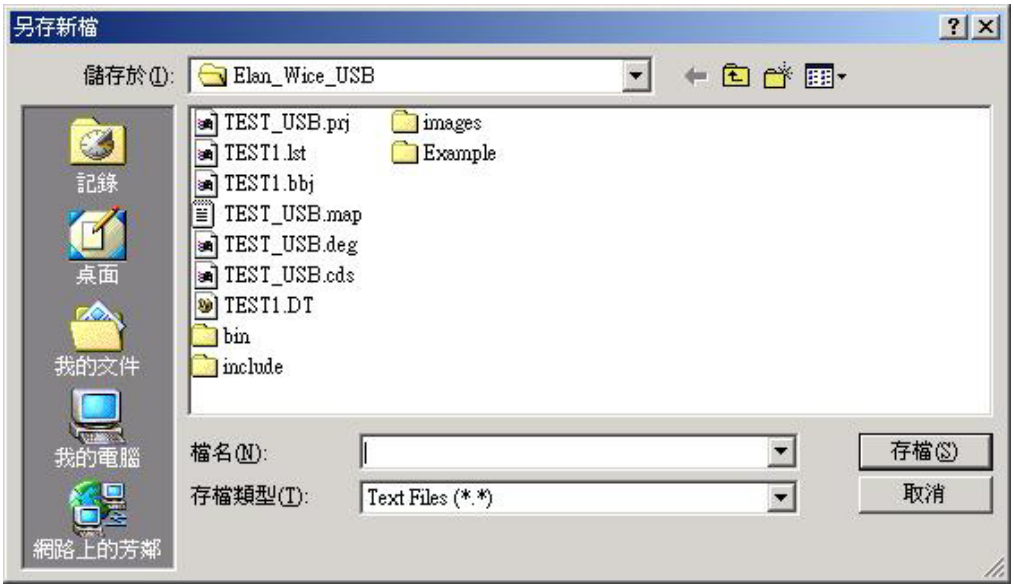


圖 36:另存新檔

Step 1: 使用滑鼠選擇欲儲存的檔案，按下滑鼠的左鍵，使選擇的視窗(window)，變成主要的視窗(Active window)。

Step 2: 選擇[File]→[Save As…]，則會顯示圖 36 另存新檔的對話盒。

Step 3: 鍵入檔案名稱，檔案名稱的編輯欄位在對話盒的[檔案名稱]的右邊。

Step 4: 使用滑鼠按下[儲存檔案]的按鈕，則所選的檔案會儲存，而螢幕會出現新的檔案。

(9) Save All (儲存目前已開啟的所有檔案及計畫)

選擇[File]→[Save All]

(10) Print (列印)

列印目前所選擇的編輯視窗。

(11) Print Setup (印表機設定)

(12) Resent Files (最近關閉的檔案)

(13) Recent Projects (最近關閉的計畫)

(14) Exit(離開)

5. 編輯功能(Edit menu)

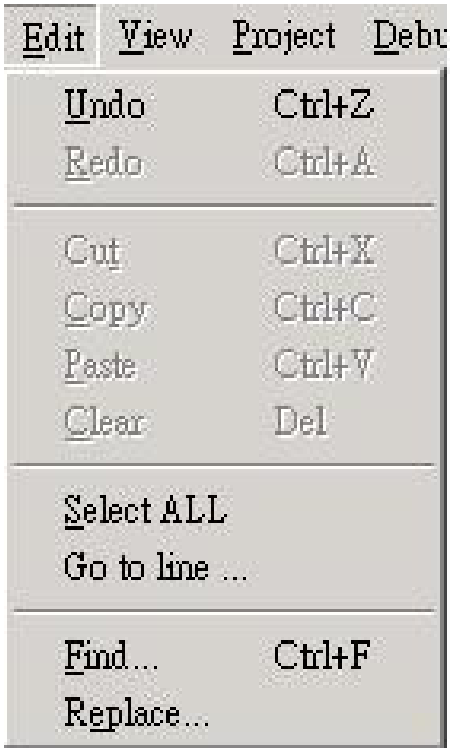


圖 37:編輯功能

(1)Undo:回復前一個編輯動作。

(2)Redo:回復 Undo 的前一個動作。

(3)Cut:剪下，將編輯視窗內反白的文字刪除。

(4)Copy:拷貝，將編輯視窗內反白的文字拷貝。

(5)Paste:貼上，將前一個拷貝的文字貼在游標所在的地方。

(6)Clear:清除。

(7)Select All:選擇編輯視窗內的所有文字。

(8)Go to line:將游標移到使用者所輸入的行數。

圖 38:輸入游標移動的行數

Step 1:輸入行數。

Step 2:按[OK]按鈕。

(9)Find:尋找使用者所輸入的文字。

圖 39:尋找字串

Step 1:輸入欲尋找的字串。

Step 2:選擇完整的字，大小寫需符合，往上或往下尋找。

Step 3:按找下一個(Find next)的按鈕。

(10) Replace:尋找並取代使用者所尋找並取代的文字。

圖 40:尋找並取代字串

Step 1:輸入欲尋找的字串。

Step 2:輸入取代後的字串。

Step 3:選擇完整的字或大小寫需符合。

Step 4:按找下一個(Find next)的按鈕。

Step 5:按取代 (Replace) 按鈕。

6. 顯示功能(View menu)

圖 41:View 的選項

(1)Project: 顯示計畫視窗(Project window)，計畫視窗中包括 Files(來源檔，副檔名為 DT 和 ASM)，References(參考檔，來源檔中的 INCLUDE 內的檔案)，List files(組譯後產生的檔案，因此每個來源檔均有一個 LIST 檔案，副檔名為 LST)，Map files(連結後產生的檔案，一個 Project 產生一個 MAP 檔案，內容主要為公用標籤(Public label)的位址。

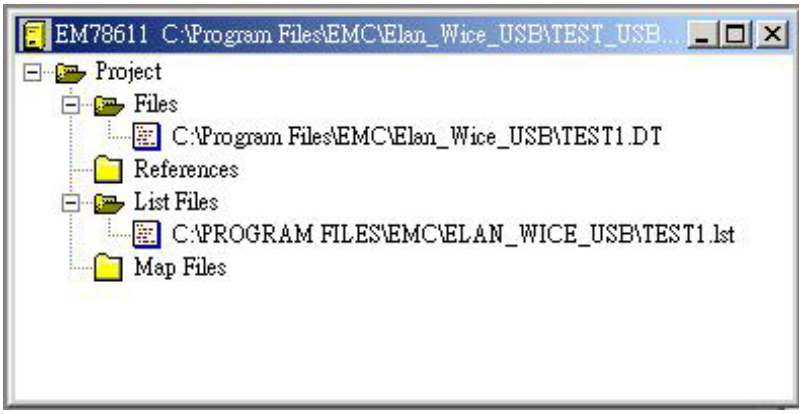


圖 42:Project 視窗

a. 增加新的檔案到計畫(Project):選擇[File]→[New…]，出現對話盒後，請選擇[File]標籤，請參照檔案功能新增部份。

b. 增加一個或多個已編輯的檔案到計畫(Project): 選擇[Project]→[Add files to project]，出現對話盒後，請選擇一個或多個檔案，請參照計畫功能增加檔案到計畫的部份。

c. 刪除計畫中的檔案:用滑鼠移到欲刪除的檔案，按下滑鼠左鍵，看到選擇的檔案反白後，按鍵盤的 Del 鍵，則可看到檔案名稱已不存在於計畫中。

(2)Special register:顯示特殊暫存器的視窗，包括累加器(Accumulator)，控制暫存器(Control)，暫存器 0 至暫存器 1F。

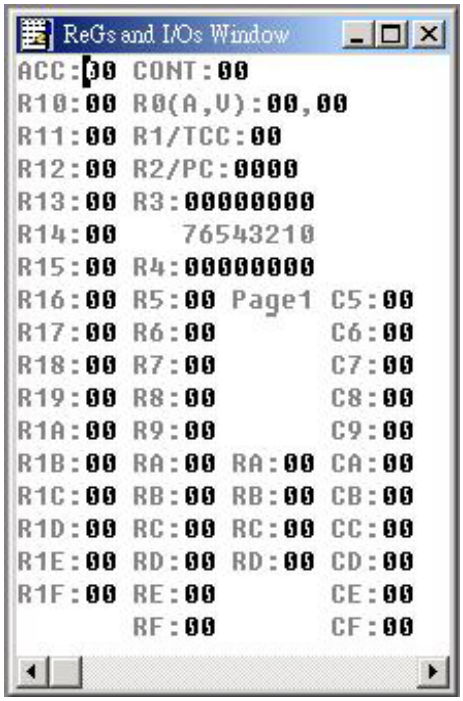


圖 43:Special register 視窗

暫存器資料修改方式:將游標移到欲修改的暫存器，若為 16 進位表示，則直接輸入 0~9，A~F，若為 2 進位表示僅能輸入 0 和 1，其餘不能輸入且 back space 及 Del 鍵均不能作用，每次修改一個數字，立即更改 ICE 暫存器的內容。

(3)RAM(Bank):顯示 RAM(BANK)一般暫存器視窗。

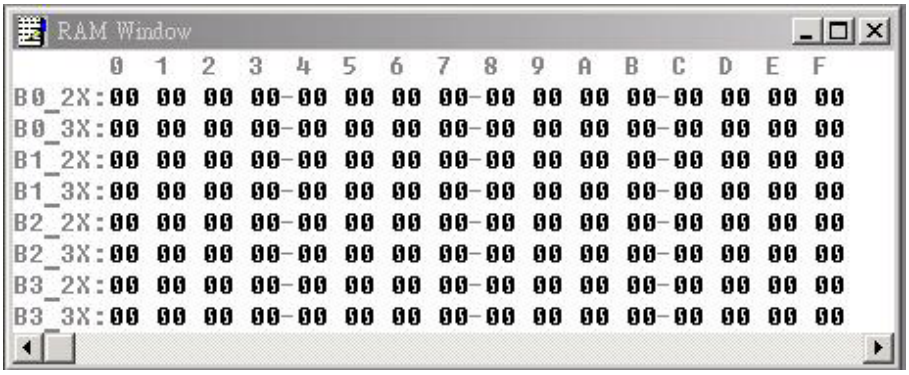


圖 44:RAM(Bank)一般暫存器視窗

暫存器修改方式如特殊暫存器的修改方式。

(4)Output:顯示一般訊息，如組譯、連結錯誤，程式曾執行過的程式碼(trace log)。

若 Output 視窗有訊息對應來源檔，則使用者使用滑鼠在訊息上，用滑鼠左鍵按雙擊(Double click)，則游標自動對應到來源檔。

(5)Watch : 顯示使用者定義變數的內容, 包括位址及值.如圖 45.

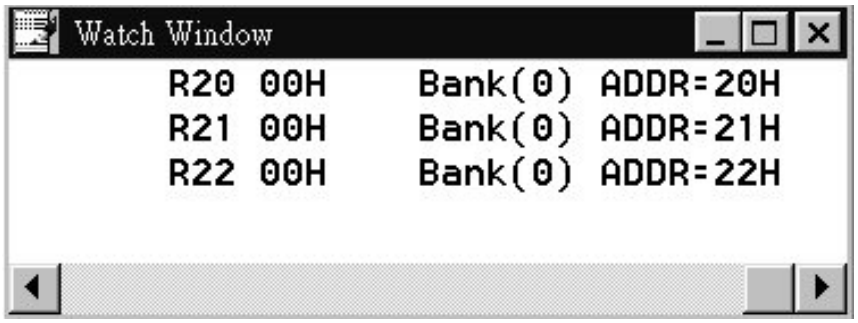


圖 45:Watch 視窗

(6)Caller RAM:目前無此功能。

(7)LCD Data:目前無此功能。

7. 計畫功能(Project menu)

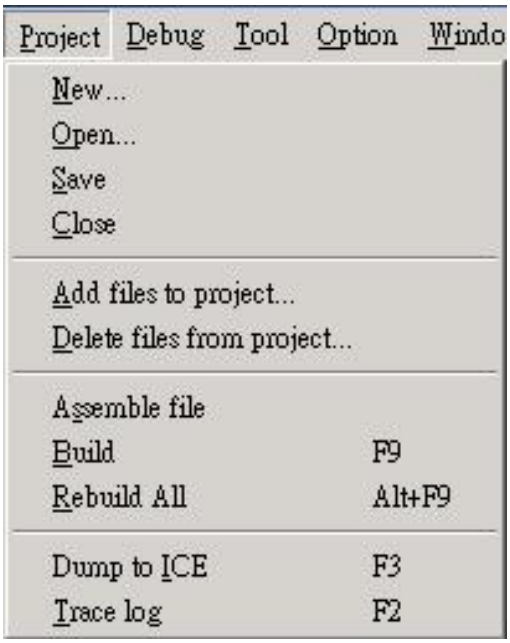


圖 46: Project 功能表

(1) New:新增一個空白的計畫(Project)。如圖 47.

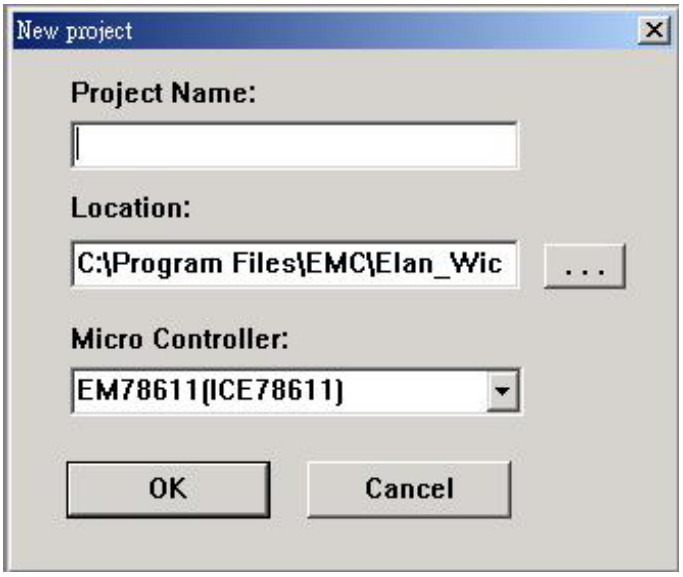


圖 47: 新增一個計畫

(2) Open:打開一個已存在的計畫(Project)。.

(3) Save:儲存一個計畫(Project)。

(4) Close:關閉一個計畫(Project)。

(5) Add files to project:增加舊檔案到計畫中(Project)，若有些檔案已經編輯，則可使用此功能將一個或多個舊檔案一次加入計畫(Project)。

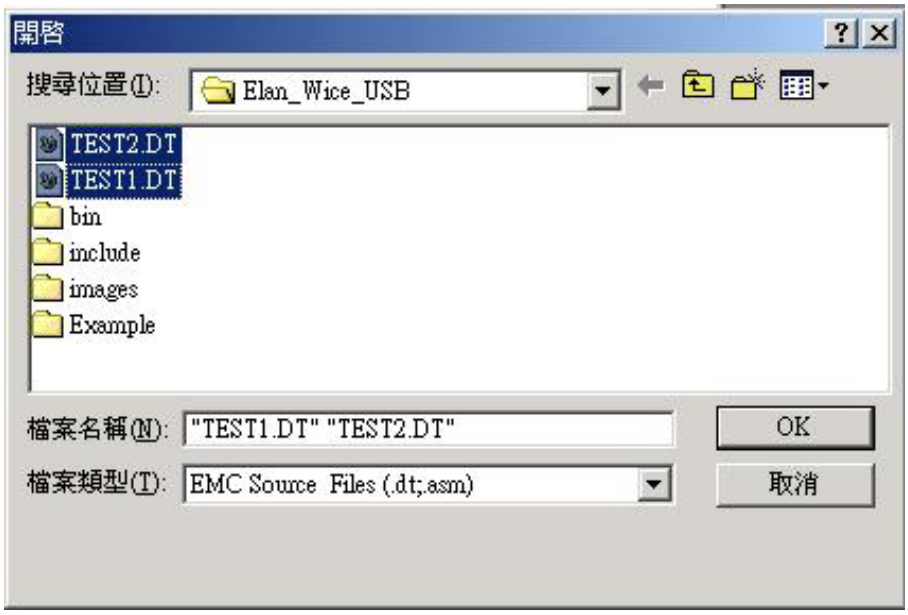


圖 48: 加入多個舊檔到 Project

(6) Delete files from project: 刪除在計畫中的檔案，如圖 49。

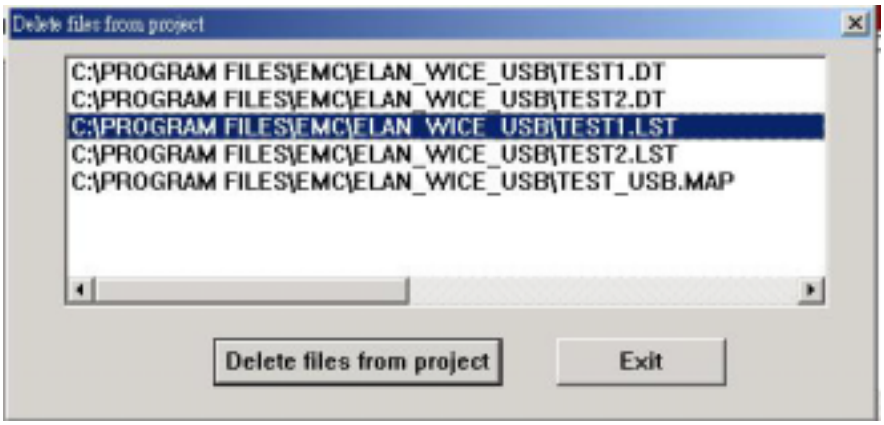


圖 49: 刪除在計畫中的檔案

(7) Assembler file:組譯檔案，組譯目前所選的編輯檔案，若有錯誤會顯示在 Output 視窗，若無錯誤會顯示 0 個錯誤，0 個警告。

(8) Build:組譯並連結檔案，組譯 Project 中所有未曾組譯或未組譯成功的來源檔，若組譯未發生錯誤，則作連結動作。組譯和連結訊息顯示於 Output 視窗。

(9) Rebuild All:組譯所有來源檔，不論來源檔是否曾經組譯過，若組譯未發生錯誤，則作連結動作。組譯和連結訊息顯示於 Output 視窗。

(10) Dump to ICE: 將程式碼載入 ICE 的程式記憶體中。會出現圖 50 載入中(Dumping)的訊息，若未載入完成，則請翻閱故障排除篇。

圖 50:載入中(Dumping)

(11) Trace log:執行 trace log 的時機，只有在執行方式為 Go、Free run、Go to cursor 及 step over (遇到 call 指令時)，才會在 Output 視窗顯示程式執行過的指令，不過最多顯示 8K 的內容。

8. 除錯功能(Debug menu)

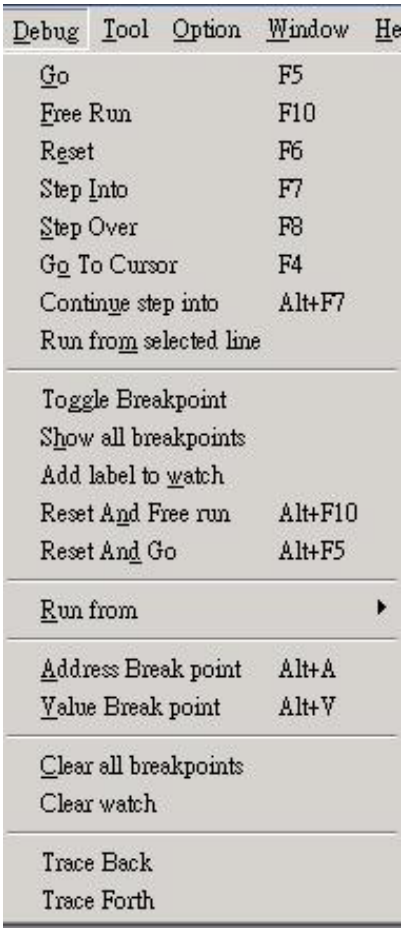


圖 51: 除錯功能

除錯功能主要是幫助使用者在每次執行完畢後，檢查暫存器所改變的值，WICE 會將有改變的暫存器值以別的顏色顯示，讓使用者容易檢查。

(1)Go(F5):F5 是熱鍵(Hot key)，Go 的執行方式是從目前的程式計數器(Program Counter)一直執行到中斷發生為止，中斷那一行的指令會執行完畢後停止執行。

(2)Free run(F10):執行方式是從目前的程式計數器(Program Counter)一直執行到使用者按圖 52 停止執行的[Cancel]按鈕為止，忽略所有已設定的中斷。

圖 52:停止執行

圖 53:此圖執行時是位於螢幕的左上角。

(3)Reset(F6):ICE 重新設定，一些特殊暫存器將會還原成如規格所提的值。如程式計數器回到原點(0 或 ROM size 最後的位址視微控制器的規格)。

(4)Step into(F7):單步執行，ICE 將會執行程式計數器所指的指令，執行一行指令即停止執行。

(5)Step over(F8):程式計數器所指的指令若為 Call 則執行完整個副程式才停止，若為其它指令，則如單步執行，一次執行一行指令。

(6)Go to cursor(F4):從程式計數器開始執行到目前游標所在的位置，忽略所有已設定的中斷點。

(7)Continue step into(Alt+F7):連續執行單步動作(step into)，直到取消執行。

(8)Run from selected line: 使用者選擇這個功能後,則下次執行的起點為游標所在的位置。

(9) Toggle breakpoint:設定中斷，設定中斷方式可參考除錯章節中的設定中斷點。

(10) Show all breakpoints: 在 output 視窗，顯示所有中斷點設定的資料。

(11) Add label to watch: 從 Watch 視窗，增加或刪除變數，如圖 54。

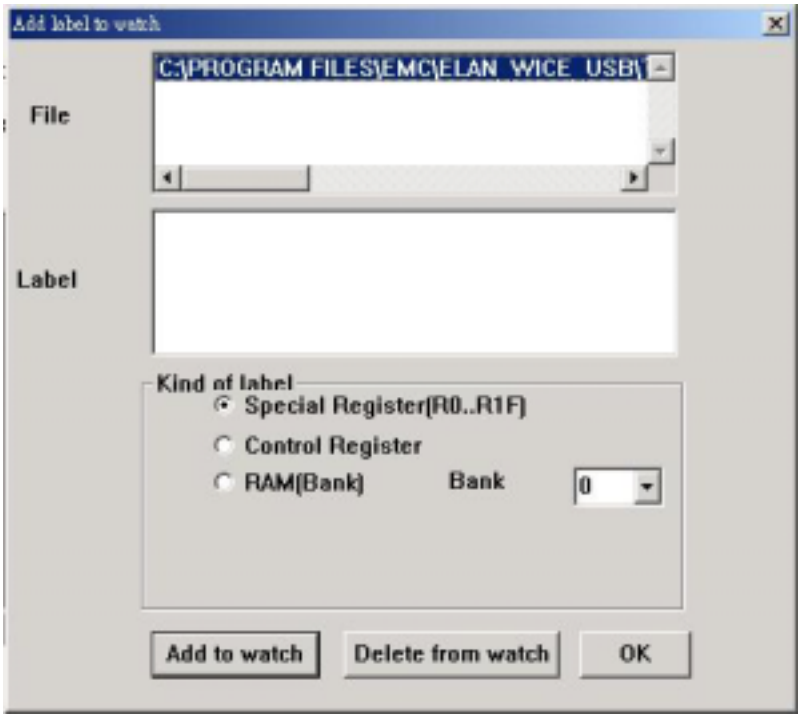


圖 54:增加 label 到 Watch 視窗

(12) Reset and Free run: ICE 重新設定並執行 Free run。

(13) Reset and Go: ICE 重新設定並執行 Go。

(14) Run from:其它執行方式，因執行起點與存放程式執行指令歷史的方式不同，分為四種。

圖 55:其它執行方式

a. Initial with 8K Step log:從程式原點開始執行直到中斷發生為止，存放中斷發生之前的 8K 執行過的指令資料。

b. Current PC with 8K Step log:從目前程式計數器開始執行，直到中斷發生為止，存放中斷發生之前的 8K 執行過的指令資料。

c. Initial with 4K-4K log: 從程式原點開始執行直到中斷發生後，再執行 4K 指令為止，存放中斷發生之前及之後各 4K 執行過的指令資料。

d. Current PC with 4K-4K log:從目前程式計數器開始執行，直到中斷發生後，再執行 4K 指令為止，存放中斷發生之前及之後各 4K 執行過的指令資料。

(15) Address break point: 使用直接鍵入程式中斷位址的設定中斷方法，如圖 56。

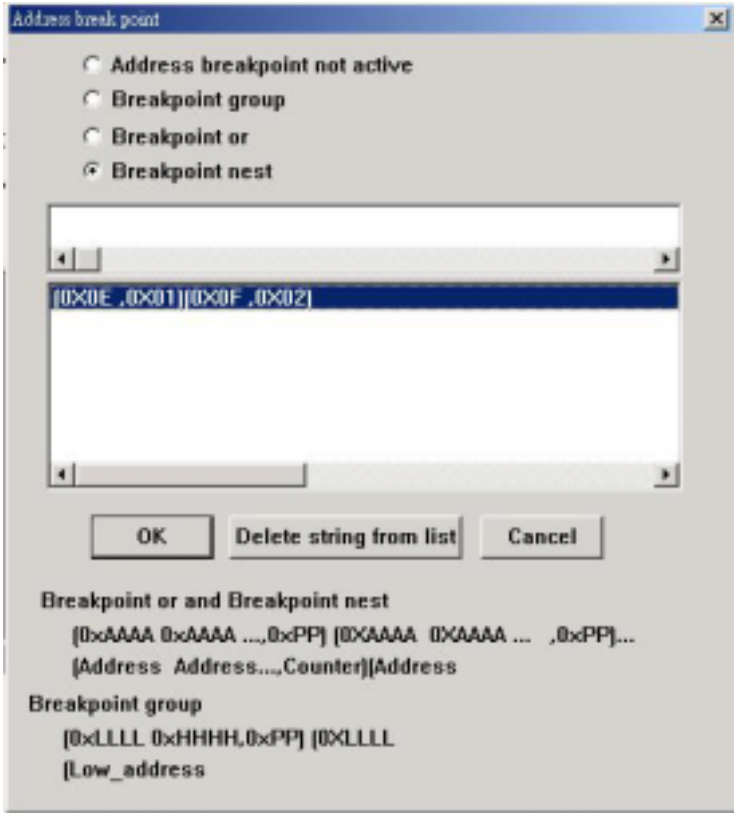


圖 56:設定位址中斷

在這對話盒所使用的中斷方式為 address 中斷, 與在來源檔上直接設中斷方式是互斥的, 也就是若使用 address 中斷,則來源中斷會無效, 因此若選擇"Address breakpoint not active", 則來源中斷方式才有作用.

Address 中斷為為三種, Group, OR, Nest, 均為互斥.而 OR 與 Nest 的設定語法一樣, 且與 Group 的設定語法不同.

(a) Group 中斷:最多 63 個 group, 每個 Group 有起始位址,終止位址及次數,當有起始及終止位址內的任何位址被執行時, 次數減 1, 假如次數等於 0, 則中斷發生,而且每個 Group 是互相獨立的.

(b) OR 中斷: 最多有 63 個 group, 每個 Group 有多個位址及一個計數器, 當 Group 內的任何位址被執行時則計數器減 1, 假如計數器等於 0 時, 則中斷發生, 而且每個 Group 是互相獨立

(c) 巢狀式中斷: 巢狀式中斷為所有設定的 Group 循序條件成立後,中斷才發生。若設

定巢狀式中斷 Group1, Group2, Group3, 則需 Group3 的計數等於 0 後, 再等待 Group2 的計數等於 0，再等待 Group1 的計數為 0 ，則中斷才發生, 因此需從最外層的 Group 的計數等於 0 後, 依序由外層到內層,若最內層的計數為 0 時，則中斷發生。

例如: 設定巢狀式中斷:

Step 1: 選擇 Breakpoint nest 。

Step 2:輸入中斷位址，假設輸入(0x03 , 0x01) (0x05 ,0x02)，則(0x03)為 group A，中斷計數 1 次(中斷計數表示程式經過這個中斷位址幾次發生中斷)，(0x05)為 group B，中斷計數 2 次。

Step 3:按[OK]按鈕。

Step 4:執行 Go(F5)指令。

Step 5:等待巢狀式中斷發生，中斷發生需先 group B 中斷條件符合，然後等待 group A 中斷條件符合後，才會發生中斷，一般巢狀式中斷用於除錯中斷程式。

9. 工具功能(Tool menu)

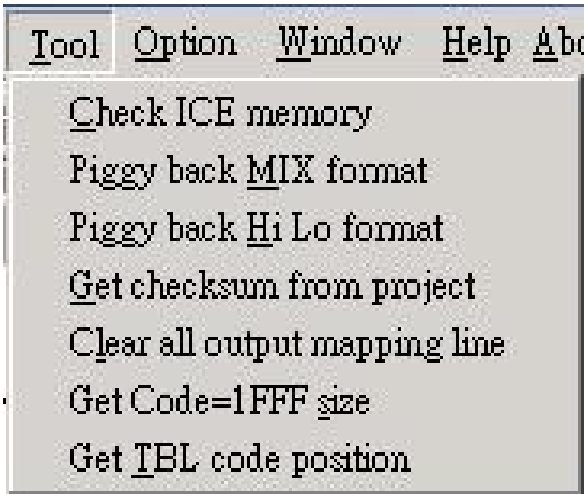


圖 57: 工具功能

(1) Check ICE memory:檢查 ICE 程式記憶體是否有壞。

(2) Piggy back MIX format:產生單顆 Piggy back 的 MIX 格式，以便於燒錄 EEPROM 之用。

(3) Piggy back Hi Lo format:產生兩顆型態的 Piggy back 的 High 及 Low 的形式，燒錄

EEPROM 之用。

(4) Get checksum from project:計算這個計畫(Project) 的 check sum，以便於下 Ordering form 之用。

(5) Clear all output mapping line:清除 Output 視窗對應到來源檔反白的光棒。

10. 選擇性的功能(Option menu)

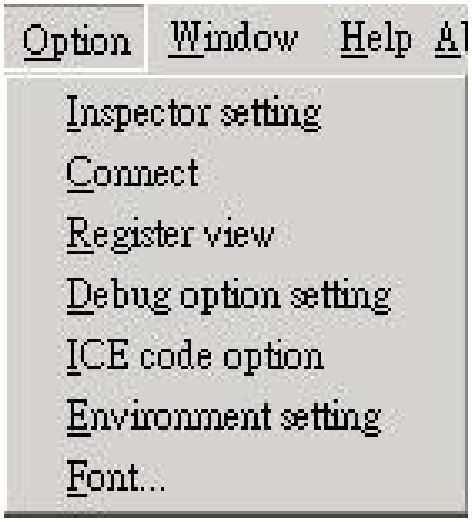


圖 58: 選擇性功能(Option)

(1)Connect:重新連接新的 ICE。可參照 WICE 環境介紹的連接(connect)功能。

(2)Register view:設定特殊暫存器顯示的型態(十六進位或二進位)。

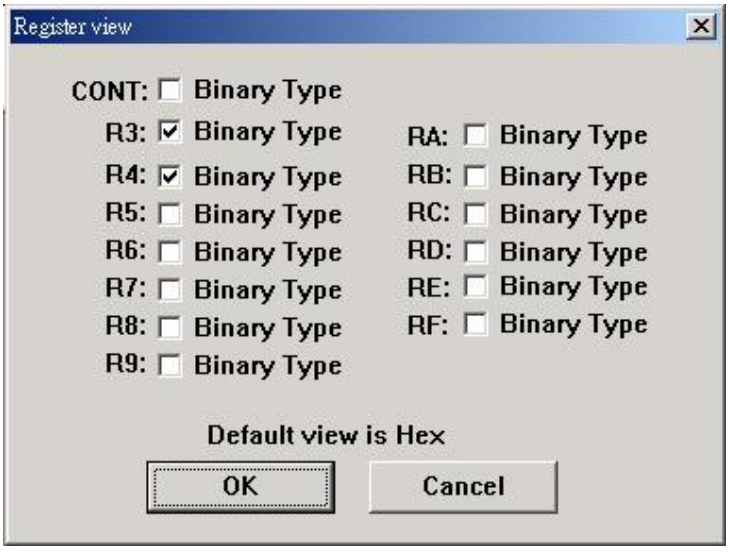


圖 59:特殊暫存器顯示型態設定(Register View)

如何設定:

Step 1:假如欲讓某些暫存器以二進位顯示，則用滑鼠移到檢查盒(Check box)上，按下滑鼠左鍵，則會出現打勾符號，若想以 16 進位顯示，則留下空白。

Step 2:按下[OK]按鈕，則特殊暫存器的顯示就如使用者所選的型態。

(3) Debug Option setting:除錯功能設定。

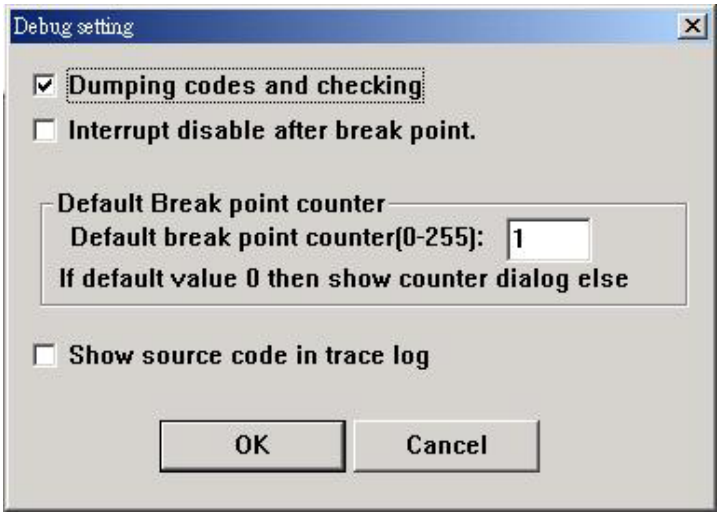


圖 60: 除錯功能設定

主要分為四區:

a. Dumping codes and checking:選擇每次將程式碼載入 ICE 時，是否檢查程式碼載入正確。

b. Interrupt disable after break point:在除錯中斷點發生後即將 ICE 的所有中斷關閉，關閉後不在打開中斷致能，除非程式有致能中斷，使用於將 TCC，COUNTER1，COUNTER2 打開的程式，在除錯中斷發生前 TCC，COUNTER1，COUNTER2 均能正常運作，為何使用本項功能，原因是 TCC，COUNTER1，COUNTER2 並不會因除錯中斷發生後，就停止，因此會造成除措器的困擾。

c. Default break point counter:預設中斷計數器，功能請參照除錯設定中斷點章節。

d. Show source code in trace log: Trace log 的內容在 output 視窗內, 預設是顯示反組譯的內容, 執行速度較快. 若選擇這個功能則將 trace log 位址對應來源檔的位址所在的那一行內容顯示在 Output 視窗, 相對的速度較慢。

(4)ICE Code option: ROM less 的功能選項，各微控制器有不同的功能選項，對話盒的功能選項顯示會依不同微控制器作調整，至於各功能請參照微控制器規格。如圖 61。

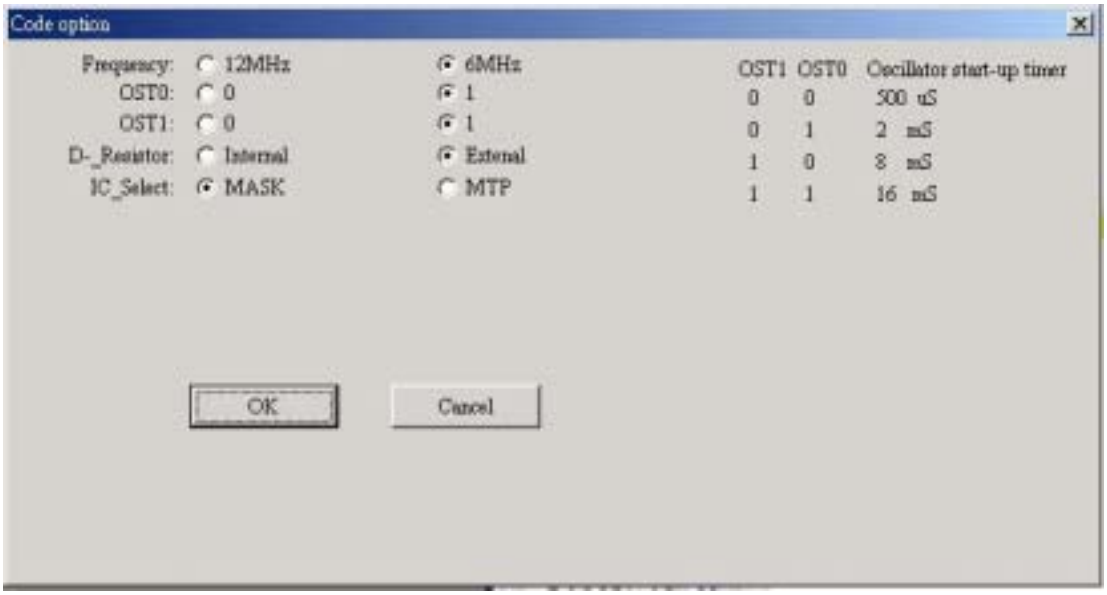


圖 61: ICE code option

(5)Environment setting:環境設定。如圖 62。

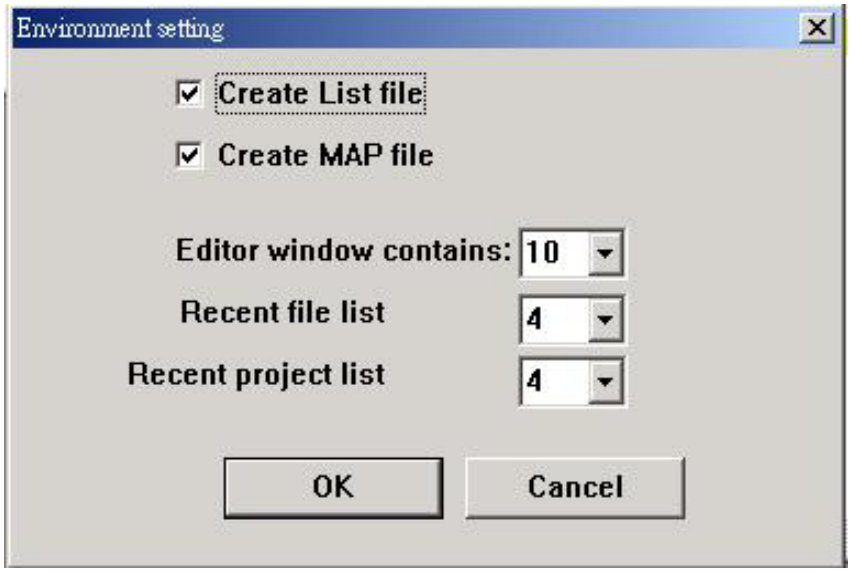


圖 62: 環境設定

a. Create List file:組譯來源檔後是否產生 LIST 檔案，LIST 檔案主要包括了行數，位址，程式碼及來源程式。

b. Create MAP file:連結後產生的 MAP 檔案，主要內容為公用標籤(public label)的位址，供程式設計者參考。

c. Editor window contains:指 WICE 的軟體執行時能同時開啟的編輯檔數目，最多 20 個編

輯檔。

d. Recent file list contains:指[File]→[Recent files]內所記錄的檔案個數，最多 10 個。

e. Recent project list contains: 指[File]→[Recent Project]內所記錄的計畫(project)個數，最多 10 個。

(6) Font:字型，可以設定編輯檔及特殊暫存器及一般暫存器的字型，所有編輯檔與 Output 視窗為同一種字型，而特殊暫存器與一般暫存器可分別設定字型，但只限於等寬字型。

11. 視窗(Window)

包含視窗排列方式及目前開啟的視窗名稱。

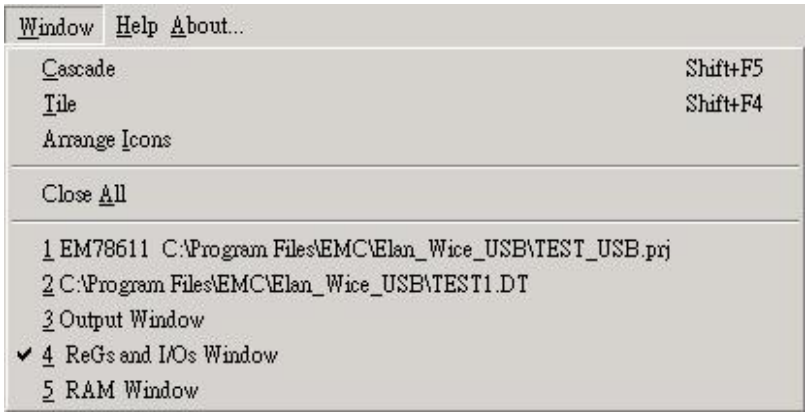


圖 63:視窗功能列

Cascade:將目前所開啟的視窗以串列方式顯示，如圖 64。

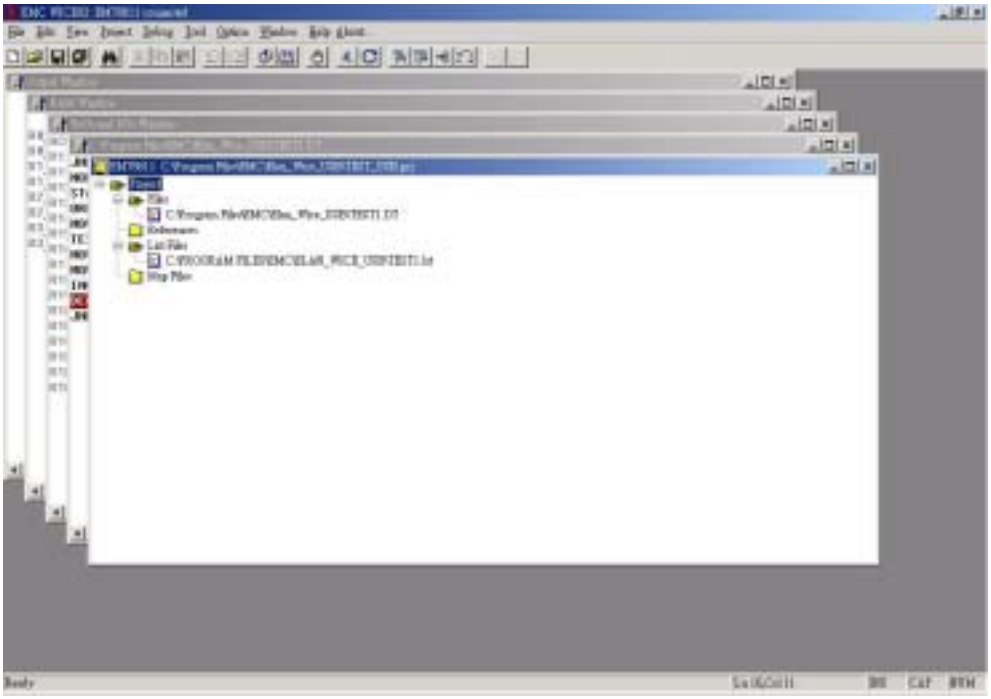


圖 64:串列方式顯示視窗

Tile:並將目前所開啟的視窗以並列方式顯示，如圖 65。

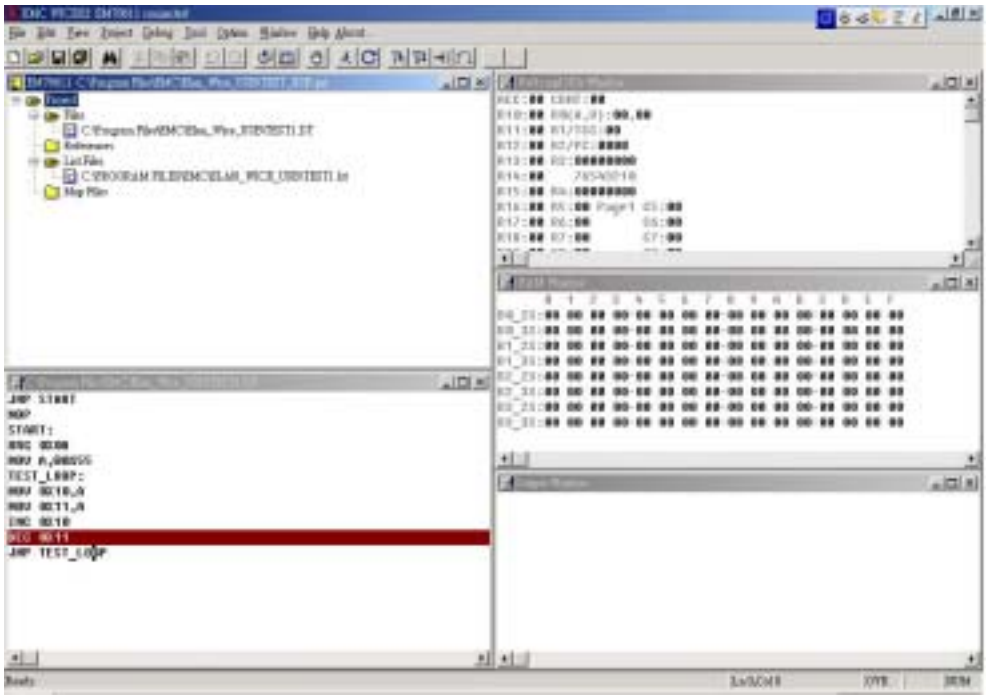


圖 65:並列方式顯示視窗

12. 求助(Help)

(1) Help 檔並未放入 WICE 程式中，但可以執行[開始] →[程式集] →[EMC WICE] →[EMC WICE ReadMe]，Read Me 檔為 HTML 格式，所以使用者電腦內只要有瀏覽器即可閱讀。

(2) About:關於本軟體的版本，WWW 伺服器及可寄信給程式設計者求助。

六、錯誤訊息(error message)

錯誤訊息主要分為五類，M 類為主程式執行時發生的錯誤，A 類為組譯時發生的錯誤，L 類為連結時發生的錯誤，D 類為除錯時發生的錯誤，U 類為 USB 連結裝置時發生的錯誤。

1. M 類: 主程式執行的錯誤

(1) "error M001: Numbers of opening editor window are the limit !!"

原因：編輯視窗數目已超過限制，因此不再產生編輯視窗。

排除：將暫時不用的編輯視窗關閉。

(2) "error M002:Not enough memory to allocate Editor Window!!"

原因：沒有足夠記憶體來產生編輯視窗。

排除：將暫時不用的編輯視窗關閉，或關閉系統上的其它應用程式。

(3) “error M003:File: [filename] is existed.”

原因：這個檔案已存在，不能產生這個檔案名稱的檔案。

排除：將檔案名稱修改後，再儲存。

(4) "error M004:File: [filename] can not be created."

原因：作業系統通知應用程式這個檔案不能產生。

排除：檢查是否硬碟滿了，系統不穩定或中毒等。

(5) "error M005:One project has been opened!!"

原因：在視窗上已有一個 Project 打開，因此不能再打開另一的 Project。

排除：先關閉已打開的 Project，再打開想使用的 Project。

(6) "error M006:Project: [filename] can not be created."

原因：作業系統通知應用程式這個 Project 不能產生，或沒有輸入 Project 的名稱。

排除：檢查是否檔案名稱沒有輸入，硬碟滿了，系統不穩定或中毒等。

(7) "error M007:The file [filename] already exists in the project."

原因：這個檔案已經加入 Project 內。

排除：不用再加入同樣的檔案到 Project 內。

(8) "error M008:File: [filename] can not be saved !!"

原因：作業系統通知應用程式這個檔案無法儲存。

排除：檢查是否檔案名稱沒有輸入，硬碟滿了，系統不穩定或中毒等。

(9) "error M009:The project: [filename] is not EMC project file format."

原因：這個 Project 不是 WICE 所規定的 Project 格式。

排除：刪除這個 Project，再產生一個新的 Project，然後加入原來檔案，再儲存。

(10) "error M010:The file: [filename] does not exist."

原因：作業系統通知應用程式這個檔案不存在。

排除：檢查這個檔案是否存在或中毒等。

(11) "error M011:The File: [filename] can not be opened."

原因：作業系統通知應用程式這個檔案不能打開。

排除：檢查這個檔案是否存在或中毒等。

(12) "error M012:The file: [filename] exceeds %dk of the max size %dk."

原因：這個檔案的長度已超過編輯視窗緩衝區所預設的大小，目前編輯視窗緩衝區的預設長度為 1M。

排除：將程式分為二個小程式，分別處理。

(13) "error M013:The copy size: %d exceed %dk of the max size %dk."

原因：拷貝的長度超過最大的長度。

排除：將拷貝的內容分成兩部份拷貝。

(14) "error M014:Memory can not be allocated."

原因：系統無法提供額外記憶體來使用。

排除：將暫時不用的編輯視窗關閉，或關閉系統上的其它應用程式。

(15) "error M015:Over 250 characters of a line !!"

原因：在這一行的文字編輯，已超過 250 個字。

排除：到下一行繼續編輯。

(16) "error M016:The active file [filename] extension name is not .dt or .asm."

原因：組譯的來源檔的副檔名須為 DT 或 ASM。

排除：將副檔名改為 DT 或 ASM。

(!7) "error M017:No file to assembler."

原因：沒有檔案可供組譯。

排除：選擇使用者所需要組譯的程式，再執行組譯。

(18) "error M018:Project file must be created."

原因：必需先打開 Project，才能作組譯及連結。

排除：打開或新增一個 Project。

(19) "error M019:The editor number is over the max editor number."

原因：編輯檔的數目已超過最大的數目。

排除：將暫時不用的編輯視窗關閉。

(20) "Error M020:No active editor window."

原因：Editor 視窗目前沒被選取。

排除：選取 Editor 視窗。

(21) "Error M021: Text field must be input characters."

原因：文字輸入欄內不能空白。

排除：輸入文字。

2. A 類: 組譯時發生的錯誤。

(1) "error A001: Can not find the [filename] file"

原因：作業系統通知應用程式這個檔案不存在。

排除：檢查這個檔案是否存在或中毒等。

(2) "error A002:main and subroutine program can not define [label name] local label !!"

原因：區域型的標籤(以$號為首的標籤)不能存在主程式及副程式。

排除：將區域型的標籤改為全域型標籤。

(3) "error A003: The syntax form should be: operation [operand][,operand]"

原因：字串語法有錯，格式需為 operation [operand][,operand]。

排除：檢查錯誤地方，重新編輯。

(4) "error A004:The label [label name] is redefined."

原因：這個標籤重覆定義。

排除：重新定義標籤名稱。

(5) "error A005: The EQU syntax is: label EQU operand"

原因：檔案中定義 EQU 的語法有錯。

排除：修改檔案中 EQU 的定義。

(6) "error A006: The INCLUDE nest deep is over 256."

原因：使用 INCLUDE 的深度已超過 256 次(INCLUDE 內有 INCLUDE)。

排除：改變程式內 INCLUDE 的方式，使深度小於 256 次。

(7) "error A007: The IF conditional expression is error."

原因：IF 的條件運算式有錯。

排除：重新編輯 IF 的條件運算式。

(8) "error A008: Attempt to divide by zero."

原因：運算式中發生除以 0 的運算。

排除：重新修改運算式。

(9) "error A009:The assembler dose not support floating point."

原因：這個組譯器並沒有支援浮點運算。

排除：移除浮點運算，改以整數運算。

(10) "error A010:The symbol [symbol name] is not defined."

原因：這個標籤沒有定義。

排除：定義這個標籤。

(11) "error A011: The macro name [macro name] is redefined."

原因：這個巨集重覆定義。

排除：重新定義巨集名稱。

(12) "error A012:The parameters name [parameter name] is the same as label."

原因：參數名稱與標籤名稱相同。

排除：重新定義參數名稱。

(13) "error A013:The parameter name [parameter name] is the same with another one."

原因：參數名稱與同一巨集定義的其它參數名稱相同。

排除：重新定義參數名稱。

(14) "error A014:The number of actual parameter dose not match with formal parameter."

原因：巨集定義的參數數目與執行巨集的參數數目不合。

排除：修改巨集執行的參數數目。

(15) "error A015:The parameter number %d does not exist."

原因：這個參數位置並沒有定義。

排除：修改巨集執行的參數。

(16) "error A016:The external symbol [symbol name] is the same name as defined label."

原因：外部定義的標籤與檔案內部定義的標籤一樣。

排除：重新定義檔案內部的標籤。

(17) "error A017:　Address of ORG　is error."

原因：ORG 位址定義有錯。

排除：重新定義 ORG 位址。

(18) "error A018:MACEXIT can not be set outside macro."

原因：MACEXIT 不能在 MACRO 以外的地方定義。

排除：移除 MACEXIT。

(19) "error A019: Parameter must be string variable."

原因：參數必須是字串變數。

排除：將參數變數定義為字串變數。

(20) "error A020:Memory can not be allocated."

原因：系統無法提供額外記憶體來使用。

排除：將暫時不用的編輯視窗關閉，或關閉系統上的其它應用程式。

(21) "error A021:The source statements exceed %d lines."

原因：來源檔的行數超過限制。

排除：將程式分為二個小程式。

(22) "error A022:The tree is error because of parser error."

原因：這一行的語法有錯。

排除：重新撰寫這一行程式。

(23) "error A023:[allocate memory type] memory fault when allocated memory."

原因：系統無法提供額外記憶體來使用。

排除：將暫時不用的編輯視窗關閉，或關閉系統上的其它應用程式。

(24) "error A024:Set assembler variable must be integer value."

原因：組譯時變數(VAR operation)的右邊的運算式的結果須是整數。

排除：修改右邊運算式。

(25) "error A025:Address of ORG must be integer value."

原因：ORG 定義的位址不是整數。

排除：重新定義 ORG 位址。

(26) "error A026:The %d address of PC is out of %d ROM size."

原因：程式計數器的值已超過 ROM 的長度，程式碼太長。

排除：將程式減少。

(27) "error A027: The assembler is exceed max pass %d."

原因：組譯的次數超過軟體的設定次數。

排除：檢查程式是否有未定義的巨集或標籤，或巨集深度太深。

(28) "error A028:The operand %d value is not including the valid data. "

原因：這個運算元的值並不是合法的值。

排除：修改運算元的運算式。

(29) "error A029:The %4X address is conflicted."

原因：程式碼的位址發生衝突，也就是有兩個程式碼位置一樣。

排除：使用[Edit]→[Find]在 LIST 檔案中尋找這個位址，然後再到來源檔修改程式位址。

(30) "error A030:The file [file name] could not be opened."

原因：作業系統通知應用程式這個檔案不能打開。

排除：檢查這個檔案是否存在或中毒等。

(31) "error A031:The configure file read error."

原因：這個微控制器描述檔讀取錯誤。

排除：重新安裝 WICE。

(32) "error A032: The file [file name] can not be opened."

原因：作業系統通知應用程式這個檔案不能打開。

排除：檢查這個檔案是否存在或中毒等。

(33) "error A033:The macro is not defined."

原因：這個巨集沒有定義。

排除：定義新的巨集，或修改巨集名稱。

(34) "error A034:The expression can not be calculated."

原因：這個運算式無法計算。

排除：修改運算式程式。

(35) "error A035:The operation [operation name] is not defined."

原因：這個運算子並沒有定義。

排除：修改這個運算子的名稱。

(36) "error A036:The PUBLIC or EXTERN label [label name]　must be address label."

原因：定義公用或外部變數，必須是位址標籤。

排除：將此標籤從公用或外部變數中移除。

(37)"error A037:The operand value can not be calculated."

原因：這個運算元的值並不能計算。

排除：修改運算元的運算式。

(37) "error A038:The symbol [symbol name] is not extern symbol."

原因：這個標籤並不是外部定義的標籤。

排除：重新定義這個標籤或 EXTERN 的宣告中定義。

(38) "error A039:The reference number is over %d limits ."

原因：PUBLIC 或 EXTERN 後的標籤數目超過最大值。

排除：將 EXTERN 的標籤分行編輯。

(39) "error A040:The length of file name [file name] exceeds 256."

原因：檔案名稱超過 256 的長度。

排除：改變檔案名稱或改變檔案目錄。

3. L 類: 連結時發生的錯誤。

(1) "error L001: Memory of [memory type] stack overflow."

原因：系統無法提供額外記憶體來使用。

排除：將暫時不用的編輯視窗關閉，或關閉系統上的其它應用程式。

(2) "error L002:The [file name] not found."

原因：作業系統通知應用程式這個檔案不能打開。

排除：檢查這個檔案是否存在或中毒等。

(3) "error L003:The Object　file format dose not belong to EMC object file format."

原因：Object 檔案不是 WICE 所定義的格式。

排除：重新組譯這個來源檔。

(4) "error L004:Symbol [symbol name] is not defined."

原因：這個標籤沒有定義。

排除：定義這個標籤。

(5) "error L005:Public symbol [symbol name] is conflict."

原因：重覆定義這個公用標籤名稱。

排除：重新定義這個標籤名稱。

(6) "error L006:ROM address %X is conflict."

原因：程式碼的位址發生衝突，也就是有兩個程式碼位置一樣。

排除：選擇[Edit]→[Find]在 LIST 檔案中尋找這個位址，然後再到來源檔修改程式位址。

(7) "error L007:The file [file name] can not be created."

原因：作業系統通知應用程式這個檔案不能產生。

排除：檢查是否硬碟滿了，系統不穩定或中毒等。

(8) "error L008:Line %d: The machine address %X exceeds ROM size %X."

原因：程式位址的值已超過 ROM 的長度，程式碼太長。

排除：將程式減少。

(9) "error L009:No project file active."

原因：沒有 Project 打開。

排除：開啟一個 Project。

(10) "error L010:No output window found."

原因：Output 視窗沒有開啟。

排除：選擇[View]→[Output]，則會出現 Output 視窗，否則請通知程式設計師。

4. D 類: 除錯時發生的錯誤。

(1) "error D001:The ICE memory is error ."

原因：ICE 記憶體發生錯誤。

排除：洽代理商。

(2) "error D002:CDS size=%XH does not match the ROM size=%XH."

原因：CDS 的程式長度與 ROM 定義的長度不一樣。

排除：重新組譯與連結，或 ICE 的編號與使用者在 WICE 選擇的微控制器的編號不同。

(3) "error D003:The project MCU type %s does not match %s ICE."

原因：計畫(Project)所選的微控制器(顯示於計畫視窗的抬頭(Title))與 WICE 所選的微控

器名稱(顯示於 WICE 主視窗的抬頭(Title))不符合。

排除：產生新的計畫或將 WICE 重新連接另一微控制器。

(4) "error D004:The line of file dose not transfer to machine address."

原因：來源檔的這一行並沒有轉成機械碼。

排除：檢查程式是否撰寫有錯誤。

(5) "error D005:The break point group number is over 64 ."

原因：設定中斷點的群組超過 64 個，如果只使用來源檔設定斷點，表示中斷點超過 64 個。

排除：移除多餘的中斷點。

(6) "error D006:The ICE is not connected to PC."

原因：ICE 並沒有連接 PC。

排除：請參照問題與討論。

(7) "error D008:The number is invalid."

原因：輸入未符合範圍的數字。

排除：重新輸入合法的數字。

(8) "error D009:The number of messages is over %d."

原因：在 Output 視窗中的訊息數目超過最大值。

排除：減少使用者定義的訊息，如 message 指令。

(9) "Warning D010:The address %04X does not match source file."

原因：程式計數器的位址無法對應到來源檔。

排除：請參照問題與討論。

(11) "Warning D011:Memory checked is error !!"

原因：檢查 ICE 記憶體有錯誤。

排除：請參照問題與討論。

(12) "error D012:Syntax error!!!"

原因：語法錯誤。

排除：修改程式為符合文法的程式。

(16)"error D013:Memory address %08X is error !!!"

原因：這個記憶體位置有錯。

排除：更換 ICE 記憶體，或恰代理商。

(17)"error D014:Cann't found breakpoint address of %s."

原因：設定來源檔中斷的地方，沒有程式位址。

排除：移除中斷點。

(18)"error D015:The number of symbols is over %d."

原因：標籤的數目超過限制。

排除：將來源檔案分成兩個或更多的檔案。

(19)"error D016:Please dump program before adding label to watch."

原因：使用[Debug]→[Add label to watch]時, 請先執行[Project] →[Dump to ICE]。

排除：執行[Project] →[Dump to ICE] 或按功能鍵 F3。

(20)"error D017:Trace log is empty "

原因：Trace log 內沒有資料。

排除：請確定執行 Go, Free run 及 Go to cursor 才有 trace log。

(21)"error D018:No trace item in trace log. "

原因：在 trace log 內找不到所要的資料。

排除：請執行別的命令。

5. U 類: USB 連接時發生的錯誤。

(1)"error U0001: Open USB Driver error."

原因：在連接畫面時，WICE USB 未在 USB 連接線上，導致無法連結。

排除：請先連接 PC 與 WICE USB 裝置後再執行 Connect。

七、問題與討論(Q&A)

1. Q：ICE 無法連接 PC?

A：請按下列步驟排除：

Step 1：檢查 ICE 與 PC 的 USB 連接線有無接好。

Step 2：檢查振盪器有無振盪，及插入方式是否正確。

Step 3：選擇[Option]→[ICE code option]設定 Code Option 選項，如頻率(Frequency)選擇是否正確為 6MHz 或 12MHz，OST(Oscillator Start-up Timer)是否選擇正確的值，如選用外掛電阻是否附掛正確，請參照微控制器的規格說明。

Step 4：如果還是無法解決，請先插拔一次 WICE USB 裝置。

Step 5：如果還是無法解決，請再換台電腦測試，以確定是否 USB Port 正常。

Step 6：重覆更換連接線，振盪器檢查是否其中一項故障。

Step 7：如果還是無法解決您的問題，請洽本公司。

2. Q：可以連接 ICE 但載入程式時，無法載入?

A：請按下列步驟排除：

Step 1：選擇[Option]→[CIE code option]設定 Code Option 選項，如頻率(Frequency)選擇是否正確為 6MHz 或 12MHz，OST(Oscillator Start-up Timer)是否選擇正確的值，如選用外掛電阻是否附掛正確，請參照微控制器的規格說明。

Step 2：請檢查 WICE 的微控制器設定與 ICE 硬體編號是否一樣。

3. Q：檢查 ICE 記憶體，發現記憶體有錯?

A：請按下列步驟排除：

Step 1：選擇[Option]→[ICE code option] 設定 Code Option 選項，如頻率(Frequency)選擇是否正確為 6MHz 或 12MHz，OST(Oscillator Start-up Timer)是否選擇正確的值，如選用外掛電阻是否附掛正確，請參照微控制器的規格說明。

Step 2：請檢查 WICE 的微控制器設定與 ICE 硬體編號是否一樣。

Step 3：如果還是無法解決您的問題，請洽本公司。

4. Q：為何一直重覆連接相同 ICE 時，執行的速度會越來越慢?

A：因為系統認為您一直無法連接 ICE，故等待時間就越來越長，讓 PC 與 ICE 的連接的機率較大。

5. Q：為何按 Step Into 無法執行?

A：當 Project 所選的微控制器名稱與系統連接的微控制器名稱不同時，所有除錯執行均無作用，直到開啟新的 Project 或連接不同的 ICE，使 Project 所選的微控制器名稱與系統連接的微控制器名稱相同時才能作用。

6. Q：將來源檔或計畫檔放在 NOVELL 網路上，為何按執行時一直作組譯動作?

A：判斷檔案是否組譯是根據來源檔，OBJ 檔，DEB 檔(除錯用)來判斷，因此您必須將 Project 檔及來源檔放在同一台電腦上，不論這台電腦在網路上或是自己的，而至於 WICE 安裝在何處則不重要。